Universidad de las Ciencias Informáticas

Facultad 6

Mercado de datos para el apoyo a la toma de decisiones sobre servidores PostgreSQL

Trabajo de diploma para optar por el título de

Ingeniero en Ciencias Informáticas

Autor: Tahimy Chaviano Cordero

Tutores: Ing. Rosnel Venero Acosta

Ing. Yoan Manuel Pérez Piñero

“Año 55 de la Revolución”

"La arcilla fundamental de nuestra obra es la juventud; en ella depositamos nuestra esperanza y la preparamos para tomar de nuestras manos la bandera."

Ernesto Che Guevara

Declaración de Autoría

DECLARACIÓN DE AUTORÍA

Declaro ser la autora de la presente tesis y reconozco a la Universidad de las Ciencias Informáticas sus derechos patrimoniales sobre la misma con carácter exclusivo.

Para que así conste firmo la presente a los días ___ del mes de junio del año 2013.

_____
Tahimy Chaviano Cordero
Firma de la Autora

_____
Ing. Rosnel Venero Acosta
Firma del Tutor

_____
Ing. Yoan Manuel Pérez Piñero
Firma del Tutor

DATOS DE CONTACTO

Tutor: Ing. Rosnel Venero Acosta

E-mail: rvacosta@uci.cu

Tutor: Ing. Yoan Manuel Pérez Piñero

E-mail: ymperez@uci.cu.

Agradecimientos:

En este momento no encuentro palabras para describir lo que siento por esas personas a las cuales les debo la vida. Gracias por el esfuerzo tan grande que hicieron para que hoy yo estuviera aquí parada, por convertirme en la persona que soy hoy. Mimi: a ti por haber vivido junto a mí los días más difíciles que pasé en esta universidad, porque siempre me diste el aliento para seguir adelante a pesar de las adversidades. Te agradezco por todo tu cariño, tus mimos, tus regaños, por compartir todo conmigo las tristezas las alegrías, por ser mi amiga. Todo estos logros te lo debo a ti, porque has sabido ser una madre excepcional...mil gracias por todo creo que nunca me cansaré de agradecerte, un beso mami. Papi: a ti porque siempre estuviste ahí para mí, por ser mi amigo, porque siempre estuviste pendiente de que no me faltara nada, por ser mi ejemplo a seguir, por guiarme por el buen camino y por tu apoyo en todos los momentos te agradezco porque nunca nunca te alejaste de mí...no tengo palabras para expresarte lo que siento por ti...papi mil gracias por haber contribuido también a este logro. A mis hermanos por ser la motivación ser hoy lo que soy y serviles de ejemplo. Maylen: a ti mi hermana querida por tu apoyo, porque siempre estuviste ahí pendiente de mí, de lo que necesitara hasta parecías mi hermana mayor, siempre te estaré muy agradecida., por ser mi confidente un beso y mil gracias. Andresito: a ti hermanito lindo porque sin saberlo fuiste mi impulso, mi fuerza para seguir, y aunque estés más grande que yo quiero que sepas que sigues siendo mi hermanito chiquito, un beso y mil gracias. Quiero agradecerle a la persona que cuida de mi mamá y siempre la mantiene contenta, a esa persona que hoy está junto a ella y que ha sido como un padre para mi hermano y para mí muchas gracias Placido por hacer que los días sean mejores para mi mamá y por cuidármela muy bien. A mi novio Daylig por su preocupación y por las tantas noches sin dormir, porque siempre estuvo pendiente de la tesis hasta parecía que era de él y por ser mi mayor apoyo en la realización de este trabajo. Por todos tus regaños porque siempre sacaste el máximo de mí en todo, por confiar en mí, por cada minuto de amor y cariño que me has brindado durante estos dos años y que espero que sean muchísimos más. A mi tía Milvia y a mi prima Maidelyn por confiar siempre en mí y apoyarme en todo. A mi abuelito Andrés por siempre quererme tanto y apoyarme en todo, por cuidar de nosotros siempre. Hoy hubiera querido que estuviera aquí pero sé que está muy orgulloso de mí. A mi abuela Gollita, a mis abuelos paternos, a mis tías y todos mis primos gracias por su apoyo. A mi otra familia Nana y Claribel porque siempre han sido mis madrinas, por acogerme como una hija más, por ser tan buenas conmigo. A la familia de Daylig por acogerme como un miembro más de ella, en especial a Michi gracias por haberme abierto las puertas de tu casa y ser tan buena persona. Quiero agradecer a mis tutores Yoan y Rosnel por su comprensión y por sus buenos consejos, por el apoyo que me brindaron en todo momento y por confiar en mí. Quiero agradecerles a los profesores que a lo largo de toda la carrera se esforzaron para formarme y prepararme para este momento: Garnache, Alexei, Kiel, Reinier, Oscar y a Jorge Luis, gracias

Agradecimientos

profe por haberme dado una segunda oportunidad. A mis amigas que siempre estuvieron ahí conmigo desde la secundaria a Claudia, Dayana, Arazay, Yanelis y a Katia. Por haber mantenido la amistad a pesar de la distancia. A lo largo de estos 6 años he conocido muchas personas especiales con las cuales hice una amistad bien fuerte agradecerles a todos los que están aquí hoy en especial a Maikel que desgraciadamente hoy no se encuentra entre nosotros por cosas de la vida, pero donde quiera que esté gracias mi amigo por tu amistad, a Rebeca, Romy, Karla por ser mis amigas desde primer año por tantos noches de estudio, por compartir mis alegrías y mis tristezas por siempre estar ahí cuando más lo necesité. A Heydi porque siempre estuvo ahí presente en el momento más difícil de mi vida, por las malas noches eso nunca lo voy a olvidar y por tantos buenos momentos que pasamos juntas. A Yasnelys por sus regaños, gracias por ser tan buena amiga, por tus consejos y por ayudarme en el formato del documento. A Yury por todas las recres que fuimos juntas y a pesar de que nuestra amistad es corta te considero una gran amiga, por tus consejos y miji por invitarme a tu casa a almorzar. A Yohana, Adalennis y a Gleibys porque además de ser buenísimas amigas ser mis otras tutoras, gracias muchachitas por sus consejos para la tesis, y porque siempre estuvieron dispuestas a ayudarme a pesar del poco tiempo que tenían. A Agustín porque a pesar de ser tan falso sigue siendo el mejor amigo que tuve en toda la universidad, gracias mi amigo por brindarme tu amistad aún casi sin conocerme, por permitirme ser tu confidente...aunque ahora ya no nos vemos mucho. Por último y no menos importante quiero agradecerle a esas otras personas que estuvieron junto a mi estos años y que son muy buenos amigos, gracias a Olivia, a Tahimé, a Vladimir, Cesar, Luis Ángel, Liandry, René, Ricardo, Yosmany, Manuel, Despaigne, al yabo, José Mario, Gema Yadira,, Dalie, Lincon, Ibet, Betty, al grupo 6508 y a los sobrevivientes a Victor, Ramiro, José, el ruso, Ariel, Ángel, Wilber, Katherine y Arianna.

Gracias a todos por estar aquí hoy.

Dedicatoria:

Quiero dedicar la tesis a mis padres por guiarme siempre por el buen camino y por confiar en mí.

A mi abuelo Andrés por todo su cariño y dedicación. Por ser mi ejemplo a seguir.

Resumen:

Con el objetivo de monitorizar el rendimiento y funcionamiento de los servidores de PostgreSQL surge en el departamento de PostgreSQL un sistema que lleva por nombre Naire. Dicho sistema está compuesto por varios componentes, uno de ellos el de reportes, que se encarga principalmente de mostrar una interfaz web donde los administradores de bases de datos sean capaces de monitorizar el rendimiento y funcionamiento de estos servidores. Para almacenar la información obtenida en este sistema existe un mercado de datos donde se realiza el proceso de análisis de esta información. Este proceso presenta algunas dificultades, pues aún no se puede realizar un análisis estadístico más detallado del aprovechamiento de los recursos de los servidores que se están monitorizando. Esta investigación surge con el objetivo de mejorar el apoyo a la toma de decisiones que realizan los especialistas del departamento de PostgreSQL a través del Centro de Tecnologías de Gestión de Datos (DATEC). El Mercado de Datos fue desarrollado utilizando la metodología propuesta por Ralph Kimball y lo planteado por el especialista en temas de almacenes de datos Leopoldo Zenaido Zepeda en su tesis de doctorado de guiar el proceso de desarrollo del software mediante los casos de uso.

Palabras claves: departamento de PostgreSQL, mercado de datos, servidores de PostgreSQL, toma de decisiones.

ÍNDICE

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

INTRODUCCIÓN

En el mundo actual las tecnologías constituyen una parte indispensable en la vida del ser humano. Teniendo en cuenta la enorme competencia que existe en el mercado internacional entre las compañías dedicadas al desarrollo tecnológico, ha surgido la necesidad de adaptarse a los constantes cambios que ocurren en este campo. Las soluciones informáticas que se obtienen mediante estas tecnologías traen consigo mejoras para el ambiente en el cual se aplican, logrando así el ahorro de tiempo, recursos y aporte de ganancias.

Actualmente es difícil concebir un área que no use, de alguna forma, el apoyo de la informática pues esta cubre desde las más simples cuestiones hogareñas hasta los más complejos cálculos científicos. En los últimos años la información se ha convertido en el eslabón fundamental para el desarrollo de cualquier sociedad, por esta razón se hizo necesario el diseño e implementación de los Sistemas Gestores de Bases de Datos (SGBD) para facilitar la gestión de grandes cantidades de información. Muchas empresas e instituciones en sus procesos de gestión de información tienen como soporte los SGBD.

Cuba, en aras de lograr un mayor desarrollo y a pesar del bloqueo impuesto por Estados Unidos hace más de 50 años, ha buscado alternativas para informatizar a todo el país. En Cuba existen hoy varios centros y empresas que se dedican al desarrollo del software. Ejemplo es la Universidad de las Ciencias Informáticas (UCI), que como pilar fundamental en el proceso tecnológico que se desarrolla en toda la nación elabora productos de software al mismo tiempo que forma profesionales en la informática.

La UCI como impulsor fundamental en la industria del software cuenta hoy con varios centros que se dedican a la producción de software en cada una de sus facultades. Ejemplo, el Centro de Tecnologías y Gestión de Datos (DATEC) perteneciente a la Facultad 6, el cual tiene como objetivo fundamental desarrollar productos y brindar servicios relacionados con base de datos y el análisis de la información. El mismo está formado por 4 departamentos, entre ellos el departamento de PostgreSQL. En dicho departamento y con el objetivo de supervisar el rendimiento y funcionamiento de los servidores de PostgreSQL, existe un sistema que lleva por nombre Naire. Este sistema cuenta con varios componentes, entre los que se encuentra el de reportes. El objetivo principal de este

componente es mostrar a los administradores de bases de datos una interfaz web donde los mismos monitorizan el rendimiento y funcionamiento de servidores PostgreSQL a partir de la evaluación de métricas. Este componente centra sus principales funcionalidades en la representación gráfica de las métricas consultadas. Hoy cuenta con funcionalidades para realizar un análisis estadístico a través del tiempo, sobre el comportamiento de los indicadores de rendimiento de los servidores que se están monitorizando. Pero aún existen otros indicadores que son de vital importancia para la monitorización de estos servidores, que en el mercado de datos para Naire 2.0 no se tuvieron en cuenta. Por ejemplo si el porciento de uso del CPU[1] es excesivamente alto esto puede indicar la necesidad de actualizar el CPU o la existencia de alguna aplicación mal optimizada o diseñada. Otros indicadores que no se tuvieron en cuenta fueron las conexiones por ip[2], el porciento de uso de la interfaz de red tanto de entrada como de salida, la cantidad de CPU que existen en el servidor, el uso de la memoria virtual y los procesos que más consumen memoria RAM[3] y CPU. Es por todo esto que no se puede obtener una información más detallada del aprovechamiento de los recursos de los servidores de PostgreSQL.

Partiendo de lo antes expuesto se define como **problema de la investigación**: ¿Cómo facilitar la toma de decisiones en la monitorización de servidores PostgreSQL?

Dicho problema tiene como **objeto de estudio:** los almacenes de datos, enmarcado en el **campo de acción:** mercado de datos para el apoyo a la toma de decisiones.

Para dar solución al problema planteado se definió el siguiente **objetivo general:** desarrollar el mercado de datos para el apoyo a la toma de decisiones sobre servidores PostgreSQL.

A partir del análisis del objetivo general se derivan los siguientes **objetivos específicos:**

✓ Fundamentar la selección de la metodología y herramientas a utilizar en el desarrollo del mercado de datos para el apoyo a la toma de decisiones sobre servidores PostgreSQL.

[1] CPU: Computer Processing Unit

[2] Ip: Internet Protocol

[3] RAM: Random Access Memory

✓ Realizar el análisis y diseño del mercado de datos para el apoyo a la toma de decisiones sobre servidores PostgreSQL.

✓ Implementar el mercado de datos para el apoyo a la toma de decisiones sobre servidores PostgreSQL.

✓ Validar el mercado de datos para el apoyo a la toma de decisiones sobre servidores PostgreSQL.

Para dar cumplimiento a los objetivos específicos se trazaron una serie de **tareas de la investigación:**

1. Caracterización de las metodologías y herramientas a utilizar en el desarrollo de almacenes de datos.

2. Levantamiento de requisitos para definir las necesidades del cliente.

3. Descripción de los casos de uso del mercado de datos para un mayor entendimiento del negocio.

4. Definición de la arquitectura del mercado de datos para especificar cada uno de los subsistemas por los que transita el mercado de datos.

5. Perfilado de datos para verificar el estado en que se encuentran los datos provenientes de la fuente.

6. Diseño del subsistema de almacenamiento donde se definen los hechos, medidas y dimensiones que van a estar contenidas en el mercado de datos.

7. Diseño del subsistema de integración para la realización del diseño de las transformaciones.

8. Diseño del subsistema de visualización donde se realizan el diseño de los cubos OLAP y de los reportes candidatos.

9. Diseño de los casos de prueba donde se definen la cantidad de casos de prueba que se van a aplicar.

10. Implementación del subsistema de almacenamiento para realizar la matriz bus, así como la elaboración del modelo de datos.

11. Implementación del subsistema de integración para la realización de la extracción, transformación y carga de los datos.

12. Implementación del subsistema de visualización para mostrar los reportes contenidos dentro de los libros de trabajos según el área de análisis identificada.

13. Aplicación de las listas de chequeo para verificar el cumplimiento de los objetivos planteados.

14. Aplicación de los casos de prueba para validar el cumplimiento de los requerimientos definidos.

15. Aplicación de las pruebas unitarias y de integración para validar que el producto cuenta con la calidad requerida.

Posible resultado:

Mercado de datos para el apoyo a la toma de decisiones de los administradores sobre los servidores de PostgreSQL a partir de las métricas de rendimiento de hardware y software.

El documento estará estructurado de la siguiente forma:

Capítulo 1: Fundamentación teórica de los Almacenes de Datos

En este capítulo se abordarán los resultados del estudio del estado del arte de la investigación, se describen los conceptos principales a tratarse durante el desarrollo del mercado de datos, así como una fundamentación de la metodología y herramientas que se tuvieron en cuenta para dar solución al problema.

Capítulo 2: Análisis y diseño del Mercado de Datos

En este capítulo se plantearán las principales características del sistema a desarrollar y se hará una descripción de la solución definiendo áreas del negocio, la arquitectura y el diseño. Se realizará un estudio del negocio donde se definen los requisitos del sistema los cuales son agrupados en casos de uso.

Capítulo 3: Implementación y pruebas del Mercado de Datos

En este capítulo se documenta todo lo referente a la implementación del mercado de datos. Se implementa el subsistema de integración de datos, la estructura de los datos, esquemas y tablas de la base de datos, así como los flujos de transformación. Se realiza la implementación del subsistema de visualización de los datos donde se diseñan los cubos OLAP y se implementan los reportes candidatos. Se valida el funcionamiento del sistema aplicando las listas de chequeo y los casos de prueba. Además de las pruebas de aceptación realizadas por el cliente de acuerdo con los requerimientos identificados por el mismo.

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

Introducción

En el presente capítulo se abordan los conceptos fundamentales relacionados con los almacenes de datos, la metodología de desarrollo que se va a utilizar y las principales herramientas para el desarrollo de la aplicación. Mediante las descripciones de estos aspectos teóricos se pretende facilitar el entendimiento de la investigación.

1.1 Sistemas Gestores de Base Datos

Un SGBD está diseñado para manejar gran volumen de información, tanto la definición de estructuras para el almacenamiento como los mecanismos para la gestión de la información. El SGBD permite a los usuarios definir, crear, mantener la BD y proporcionar un acceso controlado a la misma.
Ventajas de usar un SGBD (1):

1. Control de la redundancia: almacenamiento de los mismos datos varias veces (datos repetidos).
2. Restricción de acceso no autorizado.
3. Suministro de almacenamiento persistente de objetos y estructuras de datos de programas: datos accesibles desde otros programas y lenguajes de programación.
4. Representar vínculos complejos entre datos (relaciones).
5. Capacidad de poner restricciones de integridad
6. Suministro de múltiples interfaces de usuario.
7. Sistema de Copias de seguridad (backup) y recuperación ante fallos. Ante un fallo hay mecanismos para que la base de datos quede consistente.

1.1.1 PostgreSQL

PostgreSQL es un potente SGBD de código abierto del sistema de base de datos objeto-relacional. Cuenta con más de 20 años de desarrollo activo y una arquitectura probada que se ha ganado una sólida reputación de fiabilidad e integridad de datos. Se ejecuta en los principales sistemas operativos, incluyendo Linux, UNIX (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64) y Windows. (4)
PostgreSQL es uno de los proyectos de código abierto más grandes y maduros que existe en la actualidad. La comunidad que le da soporte no sólo ha producido un SGBD, sino que también ha

tenido en cuenta la confección de una buena documentación de código abierto disponible, cuenta además con diferentes empresas, contribuciones de proveedores comerciales y programadores de código abierto que apoyan y contribuyen de distintas maneras con el proyecto, basado en que el mismo está liberado bajo la licencia Berkeley Software Distribution (BSD) lo que permite que su código fuente sea accesible para todos sin costo alguno, con la libertad de usar, modificar y distribuir PostgreSQL.(4)

Debido a las características enunciadas anteriormente se decide utilizar PostgreSQL como SGBD por ser un gestor estable y seguro, por tener el código fuente disponible bajo los términos de la licencia de código abierto y por ser utilizado en el departamento PostgreSQL.

Servidores de Base de Datos

Los servidores de base de datos surgen con el propósito de manejar grandes volúmenes de información y poderla compartir con varios clientes de forma segura. Sirven para almacenar datos en tablas que están relacionadas entre sí y permite que varios usuarios conectados a la misma vez puedan manejar la información. Los servidores son utilizados para brindar información a los ordenadores que se conecten a él y además deben ser capaces de proveer un conjunto de aplicaciones que permitan administrar, gestionar y monitorizar el propio SGBD y a su vez facilitar servicios de mejora para el mismo, de forma fiable, rentable y con alto rendimiento. Estos se encargan de enriquecer la concepción de lo que son los SGBD, incorporando otras funcionalidades y abstracciones, permitiendo de este modo contar con un sistema que posee otras características particulares, como pueden ser la capacidad de monitorización y administración remota.

1.2 Definición de los almacenes de datos

Cuando se habla de un almacén de datos (AD) no se pueden dejar de mencionar dos personas que hicieron grandes aportes en esta área. Uno de ellos es Bill Inmon quien en términos de las características del repositorio de datos plantea que: “Un almacén de datos es una colección de datos orientados por temas, integrados, variables en el tiempo y no volátil para el apoyo de la toma de decisiones” (5).

Y Ralph Kimball plantea que: “Un almacén de datos es una copia de las transacciones de datos específicamente estructurada para la consulta y el análisis, es la unión de todos los Data marts de una entidad…" (6).

Un almacén de datos es un repositorio lógico de datos donde se agrupan diferentes tipos de información para su posterior explotación. Posee un conjunto de características de integración de datos, seguridad y análisis. Reúne la información de múltiples fuentes para luego realizar un análisis de la misma para el apoyo a la toma de decisiones.

1.2.1 Características de un almacén de datos

Según Bill Inmon las principales características de un almacén de datos son (7):

Integrado: los datos almacenados en el almacén de datos deben integrarse en una estructura consistente, por lo que las inconsistencias existentes entre los diversos sistemas operacionales deben ser eliminadas. La información suele estructurarse también en distintos niveles de detalle para adecuarse a las distintas necesidades de los usuarios.

Temático: sólo los datos necesarios para el proceso de generación del conocimiento del negocio se integran desde el entorno operacional. Los datos se organizan por temas para facilitar su acceso y entendimiento por parte de los usuarios finales. Por ejemplo, todos los datos sobre clientes pueden ser consolidados en una única tabla del almacén de datos. De esta forma las peticiones de información sobre clientes serán más fáciles de responder dado que toda la información reside en el mismo lugar.

Histórico: el tiempo es parte implícita de la información contenida en un almacén de datos. En los sistemas operacionales, los datos siempre reflejan el estado de la actividad del negocio en el momento presente. Por el contrario, la información almacenada en estos sirve entre otras cosas, para realizar análisis de tendencias. Por tanto el almacén de datos se carga con los distintos valores que toma una variable en el tiempo para permitir comparaciones.

No volátil: el almacén de información de datos existe para ser leído pero no modificado, por lo cual la información es permanente, significando la actualización del almacén de datos como la incorporación de los últimos valores que tomaron las distintas variables contenidas en él sin ningún tipo de acción sobre lo que ya existía.

1.2.2 Ventajas y desventajas de los almacenes de datos

Ventajas (8)**:**

- ✓ Integrar datos históricos sobre la actividad de la organización (o negocio) en un único repositorio.
- ✓ Analizar los datos del negocio desde la perspectiva de su evolución en el tiempo.
- ✓ Predecir tendencias de evolución del negocio.
- ✓ Permite identificar nuevas oportunidades de negocio y tomar decisiones estratégicas.
- ✓ Reducir los costes materiales y humanos en la toma de decisiones.

Desventajas (8):

- ✓ Riesgo de fracaso en la construcción del sistema, al subestimar los costes de captura y preparación de los datos.
- ✓ Riesgo de fracaso en la construcción del sistema por cambios frecuentes en los requisitos de los usuarios.

1.2.3 Mercado de datos

Los mercados de datos, son un subconjunto de datos de un almacén de datos donde se almacenan la mayoría de las actividades de análisis que se llevarán a cabo, en el entorno de la inteligencia de negocio. (9)

Kimball plantea que “Un mercado de datos es una solución que, compartiendo tecnología con el almacén de datos (pero con contenidos específicos, volumen de datos más limitado y un alcance histórico menor), permita dar soporte a una empresa pequeña, un departamento o área de negocio de una empresa grande” (10).

Se puede concluir que un mercado de datos es un almacén de datos históricos relativos a un departamento de una organización. La mayor diferencia entre los almacenes y los mercados de datos, es el ámbito de la información que contienen, debido a que en los mercados de datos es más pequeño y los datos se obtienen de un menor número de fuentes, por tanto, provoca que el tiempo de desarrollo sea menor. Son una alternativa de solución, al igual que los almacenes de datos, pues el diseño y la construcción son similares.

1.2.4 Características de los mercados de datos (11):

✓ Una estructura óptima de datos para analizar la información al detalle desde todas las perspectivas que afecten a los procesos del departamento al cual está aplicado.

✓ Se centran en los requisitos de los usuarios asociados a un departamento o área de negocio concreta.

✓ Son más sencillos a la hora de utilizarlos y comprender sus datos, debido a que la cantidad de información que contienen es mucho menor que los AD.

1.2.5 Etapas de desarrollo de un mercado de datos

Análisis y diseño

Toda fase inicial de un proyecto de gran alcance como son los almacenes de datos, debe contar con bases sólidas que le permitan avanzar hacia las otras fases de desarrollo con un mayor nivel de seguridad y organización. Para desarrollar un almacén de datos es necesario reconocer las necesidades analíticas que tiene una organización y establecer los objetivos a cumplir. Este es el primer paso a seguir, el análisis y la comprensión del nuevo entorno, el entorno analítico.

Proceso de extracción, transformación y carga (ETL)

El proceso de integración de los datos tiene como objetivo fundamental unificar los datos pertenecientes a bases de datos fuente, archivos u otros sistemas de almacenamiento. Este proceso se lleva a cabo con el fin de organizar el flujo de los datos entre diferentes sistemas en una organización y aporta los métodos y herramientas necesarias para mover datos desde múltiples fuentes a un AD, reformatearlos, limpiarlos y cargarlos en otra base de datos, Datamart.

ETL se divide a su vez en tres subprocesos.

Extracción: en esta etapa se obtiene la información desde los sistemas de origen. Las fuentes normalmente se encuentran en diferentes formatos como son base de datos relacionales o ficheros planos, además pueden incluir otras estructuras diferentes. Cuando los datos son extraídos estos se convierten a un formato listo para comenzar la transformación.

Limpieza y transformación: los datos provenientes de distintas fuentes pueden estar incoherentes, tener errores o estar incompletos. Para esto se realiza un proceso de limpieza que elimina todos estos errores e inconsistencias. Luego de que los datos se encuentran limpios y homogenizados se procede

a transformar los datos para de esta forma estandarizar los códigos, corregir los datos, eliminar registros duplicados y usar conversiones y combinaciones para generar nuevos campos.

Carga: en esta fase los datos son organizados y actualizados en la base de datos. En algunas bases de datos la información antigua se sobrescribe con nuevos datos. Los almacenes de datos mantienen un historial de los registros de manera que se pueda hacer una auditoría de los mismos y disponer de un rastro de toda la historia de un valor a lo largo del tiempo.

Si no se realiza un correcto proceso de Extracción Transformación y Carga (ETL) se pudieran obtener datos incorrectos lo que afectaría el proceso de toma de decisiones, es por eso que este proceso constituye aproximadamente un 70% del trabajo de la construcción de un AD. (12)

Inteligencia de negocio

La inteligencia de negocios es un conjunto de herramientas que permiten a través de consultas, reportes y el análisis de los datos brindar a los usuarios la información de forma sintetizada con el objetivo de facilitar la toma de decisiones. Se utiliza además para comprender, mejorar el rendimiento y reducir los costes e identificar nuevas oportunidades de negocio. Las aplicaciones de la inteligencia de negocios incluyen las actividades de apoyo a las decisiones, consulta y presentación de informes, procesamiento analítico en línea (OLAP), el análisis estadístico, el pronóstico y minería de datos.

Una definición más formal sería:

Son los procesos, tecnologías, y herramientas que se necesitan para convertir los datos en información, la información en conocimiento, y el conocimiento en planes que impulsan acciones rentables para el negocio. La inteligencia de negocios abarca el almacenamiento de datos, herramientas analíticas, contenido y gestión del conocimiento. (13)

1.2.6 Modos de almacenamiento

La tecnología de Procesamiento Analítico en línea (OLAP) es una solución empleada en el campo de la inteligencia empresarial, con el objetivo de agilizar la consulta de grandes cantidades de datos. Está basado en el modelo multidimensional de los datos y de esta forma proporciona a los usuarios una visión multidimensional de los datos. Los sistemas OLAP tienen tres variantes de implementación que difieren en varias premisas referentes a la estructura de la información en Bases de datos (BD) y en la forma de acceso a ésta. (14)

ROLAP (Procesamiento Analítico en línea Relacional), accede a los datos almacenados en un AD para proporcionar los análisis. La premisa de los sistemas ROLAP es que las capacidades OLAP se soportan mejor contra las bases de datos relacionales. Este sistema utiliza una arquitectura de tres niveles, la base de datos relacional que es la encargada del manejo, acceso y obtención de la información, el motor analítico en el nivel de aplicación es el encargado de ejecutar las consultas multidimensionales de los usuarios y se integra con niveles de presentación a través de los cuáles los usuarios realizan los análisis OLAP.

MOLAP (Procesamiento Analítico en línea Multidimensional). La arquitectura MOLAP usa base de datos multidimensionales para proporcionar el análisis, su principal premisa es que el OLAP está mejor implantado almacenando los datos multidimensionalmente. El sistema MOLAP utiliza una arquitectura de dos niveles: las bases de datos multidimensionales y el motor analítico. La base de datos multidimensional es la encargada del manejo, acceso y obtención de la información. El nivel de aplicación es el responsable de la ejecución de los requerimientos.

HOLAP (Procesamiento Analítico en línea Híbrido). Un desarrollo un poco más reciente ha sido la solución OLAP híbrida (HOLAP), la cual combina las arquitecturas ROLAP y MOLAP para brindar una solución con las mejores características de ambas: desempeño superior y gran escalabilidad. Un tipo de HOLAP mantiene los registros de detalle (los volúmenes más grandes) en la base de datos relacional, mientras que mantiene las agregaciones en un almacén MOLAP separado.

De los tres tipos de sistemas OLAP mencionados anteriormente se decidió utilizar el sistema ROLAP por las características que el mismo posee. A diferencia de MOLAP Y HOLAP, los datos en el sistema de almacenamiento de datos seleccionado son accedidos directamente desde el AD u otra fuente de datos relacional.

1.3 Metodologías para el desarrollo de un almacén de datos.

En el mundo existen disímiles metodologías que se utilizan en el desarrollo de las aplicaciones informáticas. Algunas de estas metodologías no resultan apropiadas para el caso de los almacenes de datos, por lo que se hace necesario que estos tipos de sistemas utilicen sus propias metodologías para su desarrollo. Entre las metodologías que se destacan se encuentran la Metodología de Kimball y la Metodología de Inmon, en honor a sus creadores Ralph Kimball y Bill Inmon, figuras principales en

el área de los almacenes de datos por sus aportes para el desarrollo de los mismos. La principal diferencia de estas metodologías está en la forma de enfrentar el problema.

Ralph Kimball propone dividir el mundo de inteligencia de negocios entre los hechos y las dimensiones, esta propuesta lleva a una solución completa en un período corto de tiempo. Esta metodología tiene como característica principal que es ascendente (bottom-up), plantea que por cada departamento se debe implementar un mercado de datos independiente teniendo en cuenta los temas que estén relacionados con él.

La visión de Bill Inmon se basa en un enfoque descendente (top-down), pues propone que se creen primero los almacenes de datos y luego los mercados de datos. Los mercados de datos cuando son construidos descendentemente se nutren del Data Warehouse Corporativo, convirtiéndose en un complejo empresarial de base de datos relacionales. Inmon plantea que la base para el mercado de datos debe ser la creación de una base de datos relacional levemente normalizada. Por lo que los mercados de datos se crean a partir de la arquitectura relacional de los datos corporativos.

1.3.1 Metodología propuesta por el Centro de Tecnologías Gestión de Datos (DATEC)

En el presente trabajo se propone utilizar la metodología desarrollada por la línea de soluciones de almacenes de datos del Centro de Tecnología de Gestión de Datos de la facultad 6. La principal característica de esta metodología es que cubre todas las fases del ciclo de vida por las que pasa la construcción de un almacén de datos desde el levantamiento de información inicial hasta la implementación de la herramienta del negocio. Es una metodología mixta pues reúne características de diferentes metodologías de desarrollo de proyectos de integración de datos.

Esta metodología tiene como base la metodología propuesta por Kimball en 1992, pero se adapta a las características particulares de la UCI, y lo planteado por Leopoldo Zenaido Zepeda en su tesis de doctorado, de incluir los casos de uso para guiar el proceso de desarrollo, y así lograr estar más alineados a las tendencias y normas de trabajo de la universidad. Se agrega además una etapa de prueba que permite comprobar la calidad de los productos que se desarrollan. (15)

La metodología que propone DATEC recibe por nombre “Propuesta de metodología para el desarrollo de Almacenes de Datos e Inteligencia de Negocio en DATEC “. Esta metodología cuenta con ocho fases las cuales describen cada una de las etapas por las que transcurre la construcción de un almacén de datos.

A continuación se enuncian cada una de estas fases (16):

✓ **Estudio preliminar y planeación:** Se realiza un estudio minucioso en la entidad cliente. Esto incluye un diagnóstico integral de la organización, con el fin de determinar qué es lo que se desea construir y qué condiciones existen para el desarrollo y montaje de la misma. Además se llevan a cabo las tareas de planeación del proyecto.

✓ **Requisitos:** Se realiza el proceso entrevistas al cliente para determinar los requisitos de información. Se hace un levantamiento detallado de las fuentes de datos para validar la disponibilidad de la información. Además se definen los requisitos funcionales y no funcionales de la solución y se hace el análisis de los requisitos que dan paso al diseño e implementación.

✓ **Arquitectura:** Se definen las vistas arquitectónicas de la solución, aspectos como, los subsistemas y componentes, la seguridad, la comunicación y la tecnología a utilizar.

✓ **Diseño e Implementación:** Se define el diseño de las estructuras de almacenamiento de datos, se diseñan los procesos de integración de datos como, el mapa lógico de datos, los cubos OLAP para la presentación de la información, así como el diseño gráfico de la aplicación definido por el cliente. Después se implementan cada uno de los subsistemas (repositorio de datos, integración de datos, presentación de datos).

✓ **Prueba:** Se realizan las pruebas que validan la calidad del producto, comenzando por las Pruebas de Unidad llevadas, las Pruebas de Integración y Sistema, hasta llegar a las Pruebas de Aceptación con el cliente final. Esta fase no es la única en la que se realizan pruebas durante el desarrollo del proyecto, en todas las fases hay actividades de aseguramiento de la calidad.

✓ **Despliegue:** Consta de dos etapas, despliegue piloto, donde se configuran los servidores necesarios y se instalan las herramientas según la arquitectura definida, se cargan una muestra de los datos en un ambiente controlado, con el fin de mostrarle al cliente final el sistema en funcionamiento. Una vez aceptada la solución por el cliente, se realiza la carga histórica de los datos, puede ser en el mismo entorno que el despliegue piloto u otro, todo depende de las condiciones que establezca el cliente. Además se realiza la capacitación y transferencia tecnológica de la solución a los clientes. El resultado fundamental es la solución desplegada en el entorno real y en correcto funcionamiento.

✓ **Soporte y Mantenimiento:** Comienza cuando la solución está implantada y en explotación, y se ejecuta según el contrato firmado y las condiciones de soporte establecidas. Puede realizarse a través de variados servicios, que pueden ser soporte en línea, vía telefónica, web, correo u otros y el acompañamiento al cliente. Además se realizan las tareas de mantenimiento de la aplicación tan

necesarias para este tipo de desarrollo y que garantiza el adecuado funcionamiento y crecimiento del almacén de datos.

✓ **Gestión del proyecto**: Esta fase se ejecuta a lo largo de todo el ciclo de vida del proyecto. Es aquí donde se controla, gestiona y chequea todo el desarrollo, los gastos, las utilidades, los recursos, las adquisiciones, los planes y cronogramas, entre otras actividades relacionadas con la gestión de proyectos. Esta fase es la columna vertebral del proyecto y si no se ejecuta de forma continua y correcta el proyecto puede fracasar.

De las ocho fases de la metodología propuesta por DATEC para el desarrollo de un AD solo se utilizan las cinco primeras de las presentadas por el Programa de Mejora para optar por el nivel 2 de CMMI en la UCI. Las demás son gestionadas por los especialistas del proyecto.

1.4 Herramientas y tecnologías para el desarrollo de un almacén de datos

Para la realización del mercado de datos se hace necesario el uso de varias herramientas, que facilitarán el modelado del negocio, diseño, implementación y puesta en marcha de la aplicación. A continuación en este epígrafe se muestran las herramientas necesarias a utilizar en el mercado de datos para el apoyo a la toma de decisiones sobre los servidores de PostgreSQL.

1.4.1 Herramientas de modelado

Visual Paradigm 8.0

Es una herramienta CASE (Ingeniería de Sistemas Asistida por Computadora) que emplea UML como lenguaje de modelado, y ha sido concebida para soportar el ciclo de vida del proceso de desarrollo del software mediante la representación de todo tipo de diagramas. Se diseñó para distintos usuarios interesados en la construcción de sistemas de software de forma fiable, utilizando un enfoque orientado a objetos. Sus principales características son (17):

- Disponibilidad en múltiples plataformas (Windows, Linux).
- Diseño centrado en casos de uso y enfocado al negocio que generan un software de mayor calidad.
- Uso de un lenguaje estándar común por todo el equipo de desarrollo que facilita la comunicación.

- Capacidades de ingeniería directa e inversa.
- Modelo y código que permanece sincronizado en todo el ciclo de desarrollo
- Disponibilidad de múltiples versiones, para cada necesidad.
- Licencia: gratuita y comercial.
- Soporta aplicaciones Web.
- Las imágenes y reportes generados, no son de muy buena calidad.
- Varios idiomas.

1.4.2 PostgreSQL 9.1

En la presente investigación se utilizará como gestor de base de datos PostgreSQL en su versión 9.1 debido a que es un sistema de software libre y posee un grupo de características que lo hacen un potente gestor de base de datos.

Algunas de estas características son (18):

- ✓ Replicación sincrónica para clústeres: permite replicar los datos que son agregados mediante la implementación de clústeres de servidores PostgreSQL, limitando la posibilidad de pérdidas de datos.
- ✓ Regionalización por columna para bases de datos multilingües: es la posibilidad que tienen los usuarios para configurar el lenguaje de los textos por cada columna. Incluye soporte para alrededor de 50 juegos de caracteres internacionales diferentes, incluyendo idiomas arábicos y asiáticos.
- ✓ Tablas sin log para incrementar rendimiento: proporcionan una manera de mejorar el rendimiento y el tiempo de respuesta es mucho mayor, manteniendo los datos manejados dentro de PostgreSQL y reduciendo la carga de entradas y salidas.
- ✓ Brinda confiabilidad e integridad a los datos: el PostgreSQL 9.1 puede ser ejecutado en la mayoría de los sistemas operativos más utilizados en el mundo incluyendo, Linux, varias versiones de UNIX y Windows.
- ✓ Es un producto sin costos de licencia: se convierte en una alternativa extremadamente atractiva para las empresas.
- ✓ Posee numerosas interfaces nativas de lenguajes como: C + +, Java, Net, Perl, Pyton.
- ✓ Excelente documentación.

Herramienta de administración de base de datos

PgAdmin III 1.14: es una aplicación gráfica para gestionar el gestor de bases de datos PostgreSQL. Está escrita en C++ usando la librería gráfica multiplataforma wxWidgets, lo que permite que se pueda usar en Linux, FreeBSD, Solaris, Mac OS X y Windows. Es capaz de gestionar versiones a partir de PostgreSQL 7.3 ejecutándose en cualquier plataforma. Está diseñado para responder a las necesidades de los usuarios, desde escribir consultas SQL simples hasta desarrollar bases de datos complejas. El interfaz gráfico soporta todas las características de PostgreSQL y facilita enormemente la administración. La aplicación también incluye un editor SQL con resaltado de sintaxis, un editor de código de la parte del servidor. La conexión al servidor puede hacerse mediante conexión TCP/IP, y puede encriptarse mediante SSL para mayor seguridad. (19)

1.4.3 Herramientas de ETL

Las herramientas de integración de datos están destinadas a facilitar la realización de las tareas ETL, a lo largo del tiempo han ido ganando importancia, acorde a esto, las organizaciones se han ido dando cuenta de lo costoso que puede ser el tener una mala calidad de los datos. Por tal razón se hace necesaria una herramienta que permita evaluar el nivel de calidad de los datos.

DataCleaner 3.2.1

Se va a utilizar para el perfilado de datos el DataCleaner porque es una aplicación que permite la validación y comparación de los datos. Estas actividades ayudan a administrar y supervisar la calidad de los datos con el fin de garantizar que la información sea útil y aplicable a su situación del negocio. Además porque precisamente esta versión es la que hace posible el perfilado de datos para bases de datos no SQL como Mongodb que es donde se encuentra la fuente de datos con la que se va a trabajar.

Es una herramienta de código abierto y multiplataforma que está orientada a preparar los datos para cualquier proyecto en el que se deban aplicar técnicas de Calidad de datos. Permite el perfilado de los datos, supervisando la calidad de los datos con el fin de garantizar la calidad de los mismos. Con ayuda de DataCleaner es posible crear reglas de validación para la entrada de los registros. Permite ayudar y controlar la calidad de los datos almacenados. (20)

Pentaho Data Integration 4.2.1

Es una de las herramientas utilizadas en el proceso de Extracción, Transformación y Carga (ETL). Se trata de un motor de integración de datos al que se puede acceder utilizando una interfaz gráfica de muy fácil uso para definir trabajos y transformaciones.

Con el Pentaho Data Integration se puede de una manera simple, tomar datos de una fuente (archivos locales y remotos, bases de datos y repositorio), aplicar un procesamiento a dichos datos (filtros, condiciones, cálculos, consultas), y almacenar los resultados en un destino (archivos, base de datos y repositorio). Puede funcionar sobre varias plataformas a través de un sistema que soporte Java 1.4 o una versión superior, exige como mínimo alrededor de 128 MB de RAM, brinda soporte para metadatos e incorpora operaciones de transformación. Permite operar con los campos en el flujo de datos, renombrando, calculando campos en función de otros y realizando búsquedas auxiliares en base de datos. (21)

Para el proceso ETL se va a utilizar el Pentaho Data Integration, porque posee un conjunto de características que permiten facilitar el trabajo a la hora de realizar una transformación, por ejemplo que no es necesario escribir un código que indique como realizar dicha transformación solo con seleccionar el componente adecuado es suficiente. Además permite realizar la limpieza de los datos así como su validación.

1.4.4 Herramientas BI

“Pentaho Business Intelligence Suite Enterprise Edition” (Pentaho BI) versión 3.10: proporciona al usuario final una simplicidad y una escalabilidad mejorada. “Pentaho BI” es una plataforma orientada a soluciones y centrada en procesos, que además incluye los principales componentes requeridos para implementar soluciones basadas en procesos. Sobre esta plataforma se definen las áreas de interés que poseen los usuarios para la preparación de reportes, consultas dinámicas y la realización del análisis OLAP. (22)

Esta versión de Pentaho BI ofrece un diseñador de informes que ofrece nuevas características y mejoras que hacen más fácil, para los autores de reportes, diseñar informes potentes. La interfaz limpia y unificada permite un acceso rápido a los objetos de reportes de uso común, maximizando el área de diseño y las

nuevas funcionalidades ofrecen una flexibilidad inigualable para el despliegue de la información. Permite a los autores de reportes desarrollar informes muy pulidos en 4 pasos (22):

• Selección de plantilla.
• Conectar a los datos y crear consulta.
• Diseñar y definir la agrupación de los campos del reporte.
•Formatear campos y totales.

Para la visualización de los datos se va a utilizar el Pentaho BI server porque es una aplicación que permite gestionar todos los recursos del BI. Además que proporciona a los usuarios libertades para crear contenidos nuevos.

Mondrian versión 3.0.4: es un servidor de código abierto que gestiona la comunicación entre una aplicación OLAP y la BD con la fuente. Es desarrollado en Java, Servlets y JSPs (Java Server Pages) permite ser instalado en servidores de aplicaciones como JBoss. Entre sus principales características se encuentra la facilidad para el análisis de grandes volúmenes de información almacenados en BD que soporten Java Database Connectivity (JDBC por sus siglas en inglés). Mondrian soporta el lenguaje Microsoft's Multidimensional Expressions (MDX), soporta los APIs: Java OLAP (JOLAP) y XML (Xtensible Markup Language) para el análisis de aplicaciones programadas, permite crear cubos de información, los cuales se componen de archivos XML donde se definen las dimensiones y las conexiones de los datos. Los archivos XML por lo general son complejos de realizar manualmente por lo que es común utilizar herramientas gráficas para realizar la edición de estos. (22)

Como motor de consultas OLAP se va a utilizar el servidor Mondrian pues es un motor ampliamente utilizado en los entornos de Java. Además permite realizar consultas a mercados de datos y posee una alta velocidad de respuesta.

Apache Tomcat 6.0

Apache Tomcat es un servidor HTTP (Hypertext Transfer Protocol) y un contenedor de servlets. Puede funcionar como servidor HTTP o conectado a otro servidor HTTP como Apache HTTP Server o IIS (Internet Information Services). Es software libre (licencia Apache 2.0) gestionado por la fundación Apache. Es el servidor web más utilizado para trabajar en entornos web. Tomcat puede funcionar como servidor Web por sí mismo y es usado como servidor Web independiente en entornos con alto nivel de tráfico y alta disponibilidad. (23)

Pentaho Schema Workbench 3.2.1

Esta herramienta de la suite Pentaho tiene como objetivo facilitar la tarea de diseño de cubos OLAP. Permite modelar un XML con el diseño del cubo a través de opciones lógicas y automáticas que no requieren de un manejo avanzado de este formato de archivo.

La misma permite crear, editar, actualizar y publicar esquemas OLAP para ser desplegados por aplicaciones de visualización Pentaho. También agiliza de manera considerable la construcción e implementación de este tipo de soluciones y permite mejorar los tiempos de desarrollo y despliegue en la implementación de proyectos de soluciones analíticas. (24)

1.5 Métricas para el proceso de monitorización de servidores PostgreSQL.

Para el proceso de monitorización, las métricas permiten obtener un resultado cuantitativo sobre el estado en que se encuentran los servicios de la base de datos en los servidores, así como el comportamiento de los mismos. A continuación se muestran estas métricas las cuales se van a aplicar al MD que se va a desarrollar:

✓ **Uso de la memoria virtual (swap).**

Descripción: Esta métrica se refiere al porciento de utilización de la memoria virtual ó swap.

- **Indicadores**

- Espacio libre en memoria virtual.
- Espacio en uso de la memoria virtual.
- Capacidad total de la memoria virtual.

✓ **Uso del CPU.**

Descripción: Esta métrica se refiere al porciento de uso del CPU.

- **Indicadores**

- Uso del CPU

✓ **Cantidad de conexiones por IP.**

Descripción: Esta métrica se refiere a la cantidad de conexiones que se realizan desde un IP.

- **Indicadores**

- Cantidad de conexiones

✓ **Estadísticas de interfaz de red**

Descripción: Esta métrica se refiere al porciento de ancho de banda que se usa en una conexión.

- **Indicadores:**

- Uso de ancho de banda de entrada
- Uso de ancho de banda de salida

✓ **Consumo de memoria RAM y CPU**

Descripción: Esta métrica se refiere a los procesos más consumidores de RAM y CPU

- **Indicadores**

- Procesos más consumidores de RAM y CPU
- Procesos de Postgres más consumidores de RAM y CPU

✓ **Cantidad de base de datos**

Descripción: Esta métrica se refiere a la cantidad de base de datos que existen en un servidor

- **Indicadores**

- Cantidad de base de datos

✓ **Descripción de SO**

Descripción: Esta métrica se refiere al tipo de sistema operativo que se está utilizando.

- **Indicadores**

- Sistema operativo
- Versión

✓ **Descripción CPU**

Descripción: Esta métrica se refiere a la descripción de CPU que se está utilizando.

- **Indicadores**

- Modelo
- Arquitectura
- Cantidad de CPU existente

✓ **Información del disco duro**

Descripción: Esta métrica se refiere a la velocidad de escritura y lectura, así como la velocidad de transferencia del disco duro.

- **Indicadores**

- Velocidad de escritura
- Velocidad de lectura
- Velocidad de transferencia

Las métricas descritas anteriormente son las que van a ser utilizadas en el desarrollo de la presente investigación, las cuales fueron definidas por el cliente luego de un estudio previo.

Conclusiones del capítulo

En el capítulo se enmarca una panorámica de conceptos y desarrollo de los AD, así como también las metodologías utilizadas en la actualidad para la construcción de los mismos y las herramientas necesarias para la realización del presente trabajo. Luego de concluir con el desarrollo del capítulo se arribaron a las siguientes conclusiones:

- Se seleccionó la Propuesta de metodología para el desarrollo de almacenes de datos en DATEC, la misma abarca cada una de las fases por las que trascurre un AD, permitiendo guiar de esta manera el proceso de construcción del MD para el apoyo a la toma de decisiones sobre los servidores de PostgreSQL.
- Las herramientas utilizadas permitieron brindar cada una de ellas los aportes necesarios para cada proceso del desarrollo del MD: Visual Paradigm 8.0 necesario para el diseño de los diagramas que formarán parte de la solución. PostgreSQL 9.1, así como el PgAdmin III en su versión 1.14 como herramienta de interfaz gráfica permitiendo la disponibilidad de los datos. DataCleaner 3.2.1 y Pentaho Data Integration 4.2.1 para los procesos de integración, logrando que los datos sean cargados en el MD con la calidad requerida. Para obtener los reportes como resultado final para ayudar a la toma de

decisiones se seleccionaron las herramientas: Schema Workbench 3.2.1, Apache Tomcat 6.0, Pentaho BI Server 3.10 y Mondrian OLAP Server 3.2.

CAPÍTULO 2. ANÁLISIS Y DISEÑO

Introducción

En el presente capítulo se realiza un estudio del negocio y de la organización, con el objetivo de obtener las reglas del negocio, identificar los requisitos, los casos de uso y los actores que van a interactuar con el sistema. Se define además la arquitectura para el MD y se diseñan los subsistemas de almacenamiento, integración y visualización de los datos.

2.1 Necesidades del negocio

El departamento de PostgreSQL cuenta con el sistema Naire, el mismo se encarga principalmente de monitorizar los servidores del SGBD PostgreSQL. El componente de reportes con que cuenta este sistema brinda la posibilidad a los administradores de bases de datos de monitorizar estos servidores, mediante métricas definidas que se analizan en tiempo real. Pero aún no se puede realizar un análisis estadístico sobre el rendimiento de dichos servidores, pues faltan indicadores de gran importancia a la hora de realizar dicho análisis. El mercado de datos para el apoyo a la toma de decisiones sobre los servidores PostgreSQL tiene como objetivo principal dar solución a los problemas existentes en el departamento de PostgreSQL con respecto al sistema de Naire. Es por ello que es necesario realizar una detallada identificación de las necesidades de información por parte de los especialistas.

2.2 Reglas del negocio

Las reglas del negocio describen las políticas, normas, operaciones, definiciones y restricciones presentes en una organización y que son de vital importancia para alcanzar sus objetivos. A continuación se muestran las reglas del negocio definidas para el mercado de datos:

RN1: Los indicadores para las alertas de los servidores no pueden ser cero o nulo, además tienen que ser números enteros.
RN2: Los identificadores de los indicadores no pueden estar repetidos.
RN3: Los identificadores de las dimensiones no pueden ser nulos.
RN4: Los id de los indicadores no pueden ser nulos.
RN5: Los id de las dimensiones no pueden ser nulos.

RN6: La fecha no puede ser un valor nulo.
RN7: El año de los indicadores se encuentra en el nombre del fichero.
RN8: El mes de los indicadores se encuentra en el nombre del fichero.
RN9: La semana de los indicadores se encuentra en el nombre del fichero.

2.3 Especificación de los requerimientos

2.3.1 Requerimientos de información

Los Requisitos de Información (RI) son los encargados de describir la información que se debe almacenar para satisfacer las necesidades del cliente. Durante el proceso de análisis fueron identificados los siguientes requisitos de información:

RI1: Obtener el espacio libre en la memoria virtual según el servidor, el sistema operativo y el tiempo.
RI2: Obtener espacio en uso de la memoria virtual según el servidor, el sistema operativo y el tiempo.
RI3: Obtener capacidad total de la memoria virtual según el servidor, el sistema operativo y el tiempo.
RI4: Obtener cantidad de CPU según el servidor, el sistema operativo, la descripción del CPU y el tiempo.
RI5: Obtener uso del CPU según el servidor, el sistema operativo y por tiempo.
RI6: Obtener la cantidad de conexiones por ip según el servidor, el sistema operativo y el tiempo.
RI7: Obtener utilización de ancho de banda de salida según el servidor, el sistema operativo y el tiempo.
RI8: Obtener utilización de ancho de banda de entrada según el servidor, el sistema operativo y el tiempo.
RI9: Obtener la cantidad de base de datos según el servidor, el sistema operativo y el tiempo.
RI10: Obtener los procesos más consumidores de CPU según el servidor, el tipo de proceso y el tiempo.
RI11: Obtener los procesos más consumidores de RAM según el servidor, el tipo de proceso y el tiempo.
RI12: Obtener los procesos de PostgreSQL más consumidores de RAM según el servidor, el tipo de proceso y el tiempo.
RI13: Obtener la velocidad de escritura en Kb/s según el servidor, el sistema operativo y el tiempo.
RI14: Obtener la velocidad de lectura en Kb/s según el servidor, el sistema operativo y el tiempo.
RI15: Obtener la velocidad de transferencias en Kb/s según el servidor, el sistema operativo y el tiempo.

2.3.2 Requerimientos funcionales

Los Requisitos Funcionales (RF) son las capacidades o condiciones que el sistema debe cumplir. Es necesaria la identificación de los mismos para satisfacer las necesidades del cliente. A continuación se enumeran las funcionalidades que debe poseer el mercado de datos.

RF 1: Adicionar roles
RF 2: Eliminar roles
RF 3: Visualizar roles
RF 4: Modificar roles
RF 5: Eliminar usuario
RF 6: Visualizar usuario
RF 7: Modificar usuario
RF 8: Adicionar usuario
RF 9: Adicionar reporte
RF 10: Visualizar reporte
RF 11: Eliminar reporte
RF 12: Modificar reporte
RF 13: Extraer datos de la fuente
RF 14: Realizar transformación de los datos
RF 15: Cargar los datos al MD
RF 16: Personalizar reporte
RF 17: Exportar a otro formato
RF 18: Validar usuario y contraseña

2.3.3 Requerimientos no funcionales

Los Requisitos No Funcionales (RNF) son propiedades o cualidades que el producto debe tener, que le reportan al cliente confianza y seguridad en la aplicación.

Usabilidad

RNF1: Cumplir con las pautas de diseño de las interfaces.

El sistema debe tener una interfaz gráfica uniforme que incluya pantallas, menús y opciones. Las pautas de diseño se realizarán siguiendo la arquitectura de información definida.

RNF2: Diseñar un reporte del AD de manera sencilla y ágil.

Un usuario con conocimientos básicos del sistema podrá diseñar un reporte del Almacén de Datos de manera ágil y sencilla sin necesidad de ser un experto en las herramientas requeridas para ello.

Confiabilidad

RNF3: Asegurar la disponibilidad del sistema.

El sistema debe estar disponible durante el horario de trabajo. En caso de fallo, la recuperación del servicio no deberá ser de un período de tiempo muy prolongado.

RNF4: Asegurar la recuperación ante un fallo.

El sistema debe ser capaz de recuperarse ante un fallo, teniendo en cuenta la complejidad y naturaleza de éste. El tiempo para su correcta recuperación fluctúa entre 10 minutos y 72 horas. Este tiempo comprende la solución al problema, así como su validación y prueba.

Fiabilidad

RNF5: Garantizar la persistencia de la información.

Para garantizar la persistencia de la información se realizará un respaldo total de los datos del Almacén de Datos con una frecuencia diaria. Toda esta información se almacenará en el área del departamento de PostgreSQL.

Eficiencia

RNF6: Utilización de recursos.

El tiempo de respuesta será como máximo de 3 minutos. El sistema deberá permitir al menos 5 usuarios conectados sincrónicamente sin que se afecte el tiempo de respuesta.

Restricciones de diseño

RNF7: Utilizar los lenguajes de programación definidos durante la investigación.

Como lenguaje dentro del SGBD para la programación en el almacén de datos se utilizará PL/pgSQL. En la implementación de los procesos de integración de datos se utilizará el lenguaje JavaScript y también se hará uso del lenguaje MDX para realizar las consultas.

2.4 Diagrama de casos de uso, casos de uso de información, casos de uso funcionales. Descripción.

2.4.1 Diagrama de casos de uso

En el diagrama de casos de uso (ver Figura 1) se muestra la relación que existe entre los actores y los casos de uso del sistema (CUS). Especifican la funcionalidad y el comportamiento de este a través de su interacción tanto con los usuarios como con otros sistemas. A continuación se muestra el diagrama de la presente investigación:

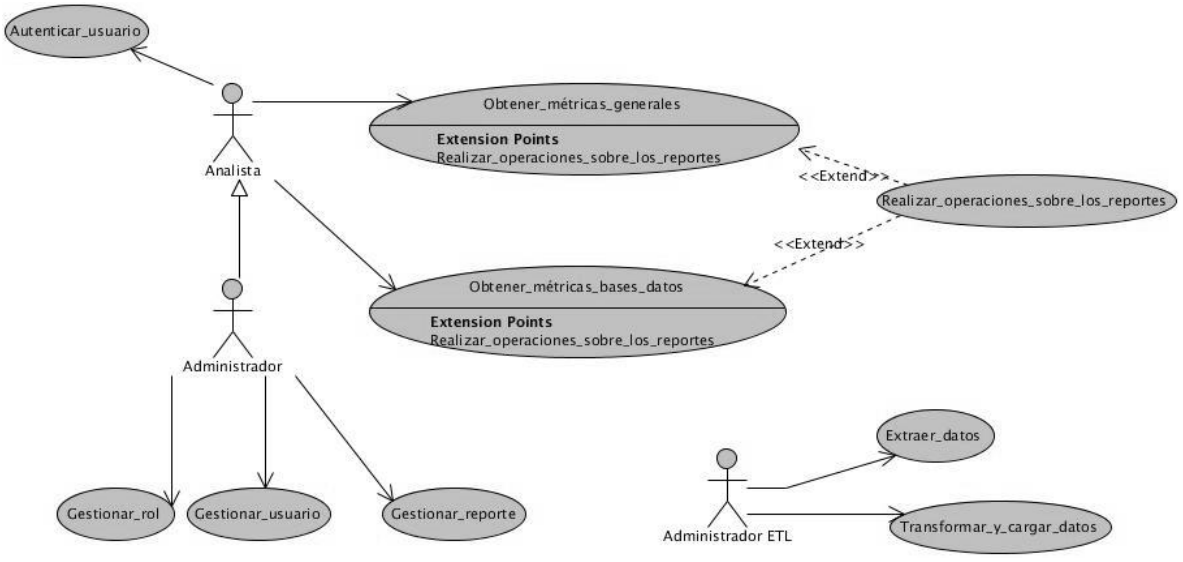


Figura 1: Diagrama de casos de uso del sistema

Para el diseño del diagrama de casos de uso se tuvieron en cuenta los siguientes patrones de casos de uso:

- Crud total: este patrón permite modelar las distintas operaciones que trae consigo el gestionar una entidad, por ejemplo gestionar rol, esto brinda la posibilidad al administrador de eliminar, modificar e insertar un rol.
- Múltiples actores: es cuando más de dos actores pueden inicializar un mismo caso de uso, es decir cuando tienen un rol común, ejemplo el administrador que hereda del analista.
- Concordancia de adición: este patrón se basa principalmente en extender los casos de uso compartiendo las secuencias de acciones, ejemplo el caso de uso realizar operaciones sobre los reportes que extiende de los casos de uso obtener métricas generales y obtener métricas de bases de datos.

A continuación la Tabla 1 muestra la descripción de cada unos de los actores que interactúan con el sistema:

Tabla 1: Actores del sistema

Actor	Objetivo
Administrador	Responsable de administrar los usuarios del sistema, asignarles sus respectivos roles, además de administrar dichos roles y los reportes.
Analista	Es el encargado de consultar y analizar la información de los diferentes indicadores
Administrador ETL	Se encarga de realizar la extracción, transformación y carga de los datos

2.4.2 Casos de uso de información

CU1_Obtener datos del análisis del rendimiento de los servidores: se refiere a la visualización de los reportes de los indicadores para el análisis del rendimiento de los servidores de PostgreSQL

CU2_Obtener datos de la descripción de CPU: visualiza los reportes de los indicadores para el análisis de los datos referentes a la descripción del CPU existente en el sistema.

CU3_Obtener datos de los procesos más consumidores: se refiere a la visualización de los reportes de los indicadores de los diferentes tipos de procesos más consumidores de RAM y CPU existentes en el sistema.

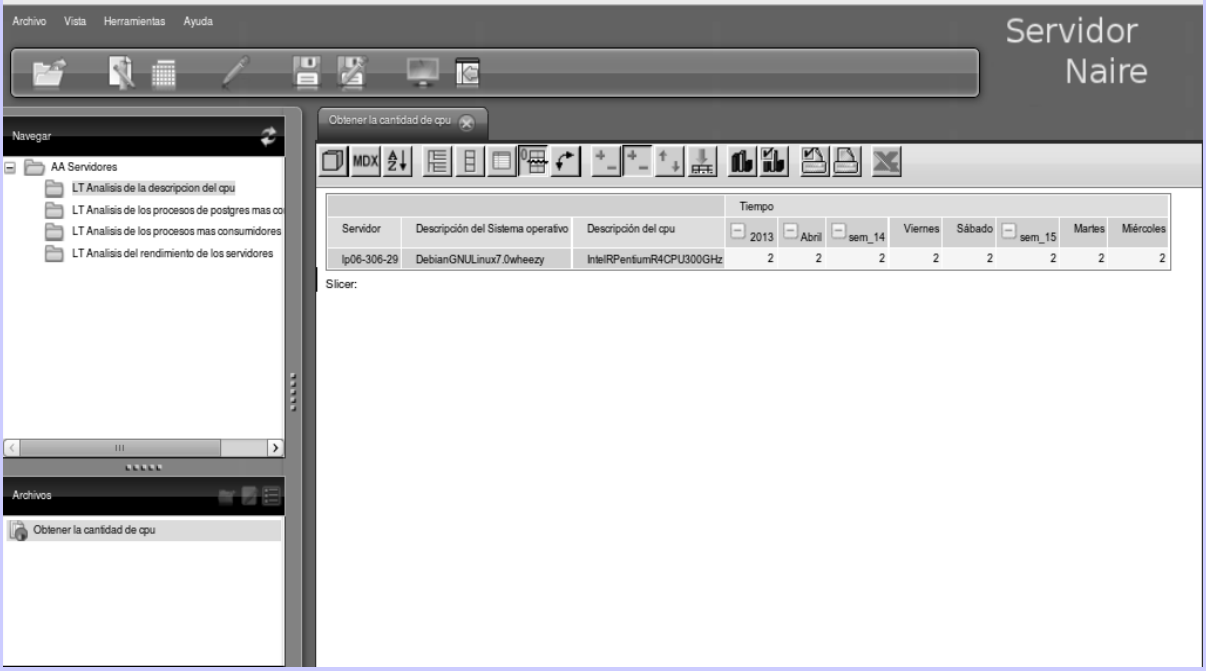
CU4_Obtener datos de los procesos de PostgreSQL más consumidores: se refiere a la visualización de los reportes de los indicadores de los diferentes tipos de procesos de PostgreSQL más consumidores de RAM y CPU.

A continuación en la Tabla 2 se muestra textualmente la descripción del CU1 Obtener datos del análisis del rendimiento de los servidores. El resto de las descripciones se encuentran en el artefacto "Especificación de casos de uso". (Ver expediente de proyecto):

Tabla 2 CU 1 Obtener datos del análisis del rendimiento de los servidores

Objetivo	Presentar reportes relacionado con el rendimiento de los servidores
Actores	Analista: (Inicia) validando los datos de un usuario en el sistema
Resumen	El caso de uso inicia cuando el analista desea consultar los datos referentes al análisis del rendimiento de los servidores y finaliza cuando el usuario visualiza el reporte.
Complejidad	Media
Prioridad	Media
Precondiciones	Analista autenticado, completitud del mercado de datos y carga de los datos.
Postcondiciones	Disponibilidad de opciones (cruce de variables) de reportes relacionados con el CU Obtener los datos del análisis del rendimiento de los servidores
Flujo de eventos	

Flujo básico <Presentar reportes relacionado con el rendimiento de los servidores >		
	Actor	**Sistema**
1.	El actor accede a la aplicación.	
2.		El sistema muestra la interfaz principal con las áreas de análisis.
3.	El actor selecciona el área de análisis A.A Servidores.	
4.		El sistema muestra los libros de trabajo que están contenidos dentro del A.A Servidores.
5.	El actor selecciona el libro de trabajo L.T Análisis del rendimiento de los servidores.	
6.		El sistema muestra los reportes contenidos dentro del L.T Análisis del rendimiento de los servidores.
7.	El actor selecciona el reporte deseado.	
8.		El sistema muestra el contenido del reporte dentro del área de trabajo y brinda opciones al actor de realizar cambios en el reporte para su mejor análisis. Ir al caso de uso **Realizar operaciones sobre los reportes**. Finaliza el caso de uso.
Prototipo de interfaz		



Opciones de reportes Autenticar_usuario

Perspectivas de análisis	**Posibles resultados**	
	Medidas	**Periodicidad**
Variables de entrada relacionadas con el CU Obtener los datos del análisis del rendimiento de los servidores: • Día • Semana • Mes • Año • Servidor • Sistema Operativo	Variables de salida disponibles en el LT Análisis del rendimiento de los servidores: • espacio_libre_swap • espacio_uso_swap • capacidad_total_swap • uso_cpu • conexiones_por_ip • uso_ancho_banda_entrada • uso_ancho_banda_salidad • cant_base_datos • veloc_escritura	Rango de tiempo en que se solicitan las variables de salida: • Anual • Mensual • Semanal • Diario

		• veloc_lectura • veloc_transferencia	
Relaciones	**CU Incluidos**	No aplica	
	CU Extendidos	CU Realizar operaciones sobre los reportes	
Requisitos no funcionales	Sección: "3.2 Requisitos no funcionales" del documento: "0114_Especificación de requisitos de software".		
Asuntos pendientes	-		

2.4.3 Casos de uso funcionales

A continuación en la Tabla 3 se muestra la descripción de los casos de uso funcionales identificados en el negocio:

Tabla 3 : Descripción de los casos de uso funcionales

Casos de uso	Descripción
Obtener métricas generales	Visualiza los reportes relacionados con los indicadores del rendimiento de los servidores y la descripción de CPU
Obtener métricas de base de datos	Visualiza los reportes relacionados con los indicadores presentes en los procesos más consumidores ya sean de PostgreSQL o del sistema.
Transformación y carga de los datos	Realiza la transformación de los datos y luego realiza la carga de los mismos en el mercado de datos.

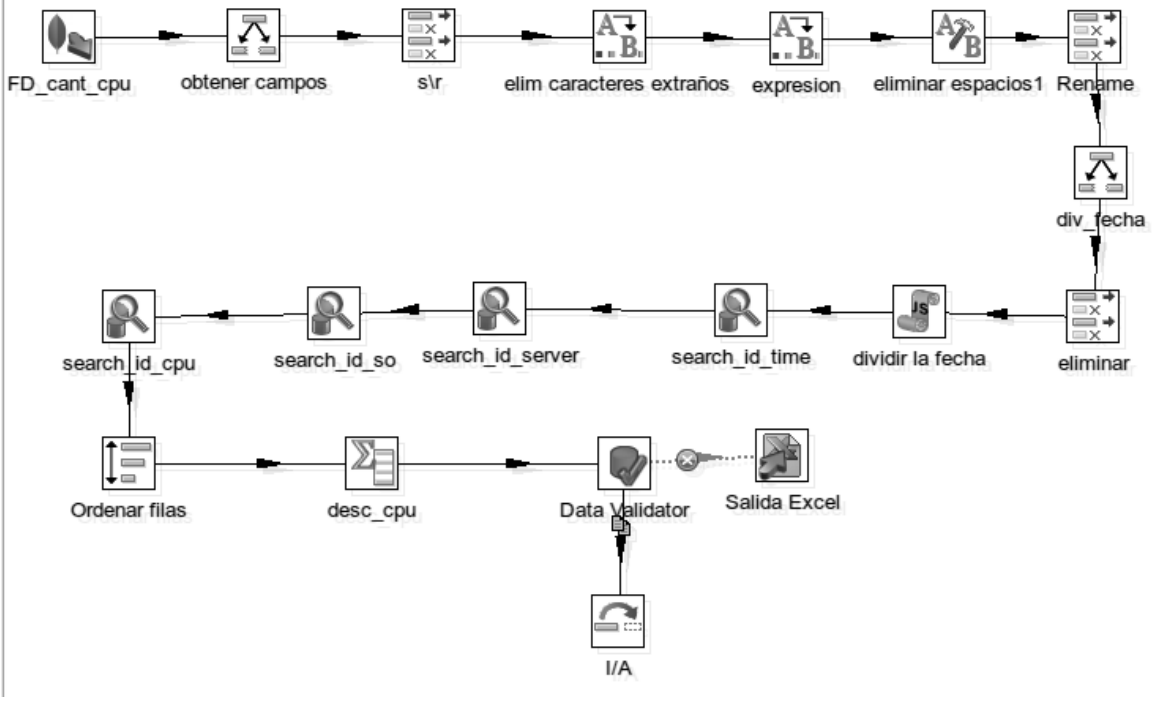
Extraer datos	Realiza la extracción de los datos provenientes de la fuente.
Administrar rol	Inserta y elimina los roles.
Administrar usuario	Inserta y elimina los usuarios que interactúan con el sistema.
Autenticar usuario	Realiza la autenticación de los usuarios en el sistema.
Realizar operaciones sobre los reportes	Visualiza los reportes y presenta funcionalidades para su configuración.

A continuación en la Tabla 4 se muestra la descripción textual del CUF4. Extraer datos. El resto de las descripciones textuales se encuentran en el artefacto Especificación de casos de uso. (Ver Expediente de Proyecto).

Tabla 4: Caso de uso Extraer datos

Caso de Uso	Extraer datos
Actores	Administrador de ETL
Resumen	El CU inicia cuando el actor desea realizar la extracción de los datos correspondientes a las fuentes de información. Se extraen los datos de la fuente. El CU finaliza una vez que los datos seleccionados por el actor son extraídos.
Complejidad	Media
Prioridad	Media
Precondiciones	Disponibilidad de las fuentes

Poscondiciones	Los datos seleccionados de las fuentes de información quedan extraídos y disponibles para transformar.	
Flujo de eventos		
Flujo básico Extraer datos		
	Actor	**Sistema**
1.	El actor interactúa con la herramienta Pentaho Data Integration (PDI) para realizar la extracción de los datos.	
1.1		El sistema muestra el área de trabajo y le da al actor la posibilidad de cargar la transformación.
2.	Selecciona la transformación a cargar o realiza los pasos para una nueva transformación.	
3.	Configura los parámetros de entrada de la transformación.	
3.1		El sistema almacena los datos en el área temporal.
4	El actor previsualiza los datos.	
4.1		El sistema muestra los datos existentes en esta área.
5	El actor realiza la acción de aceptar y finaliza.	
Prototipo de interfaz		



Flujos Alternos		
	Actor	**Sistema**
4.1		El sistema muestra un mensaje de error.
2	El actor vuelve al paso 3.	

Relaciones	**CU Incluidos**	No aplica.
	CU Extendidos	No aplica.
Asuntos pendientes	[Posibles mejoras al caso de uso.]	

2.5 Definición de la arquitectura base del mercado de datos

La arquitectura del mercado de datos ofrece un entorno de análisis, monitoreo y control de la información existente en el sistema Naire sobre los servidores de PostgreSQL, esta posee una estructura de tres subsistemas los cuales se comunican a través de los protocolos HTTP y TCP (Protocolo de Control de Transmisión)/IP (Protocolo de Internet):

- **Subsistema de integración:** es el encargado de realizar todo el proceso de limpieza, carga, extracción e integración de los datos provenientes de la fuente ya sea en Excel o base de datos.
- **Subsistema de almacenamiento:** contiene los esquemas y dentro de estos los hechos y dimensiones identificados en el mercado de datos.
- **Subsistema de visualización:** se encarga principalmente de la visualización mediante gráficos de la información almacenada agrupándolas en libro de trabajo que se corresponden con el área de análisis identificada.

A continuación en la Figura 2 se muestra la arquitectura del mercado de datos:

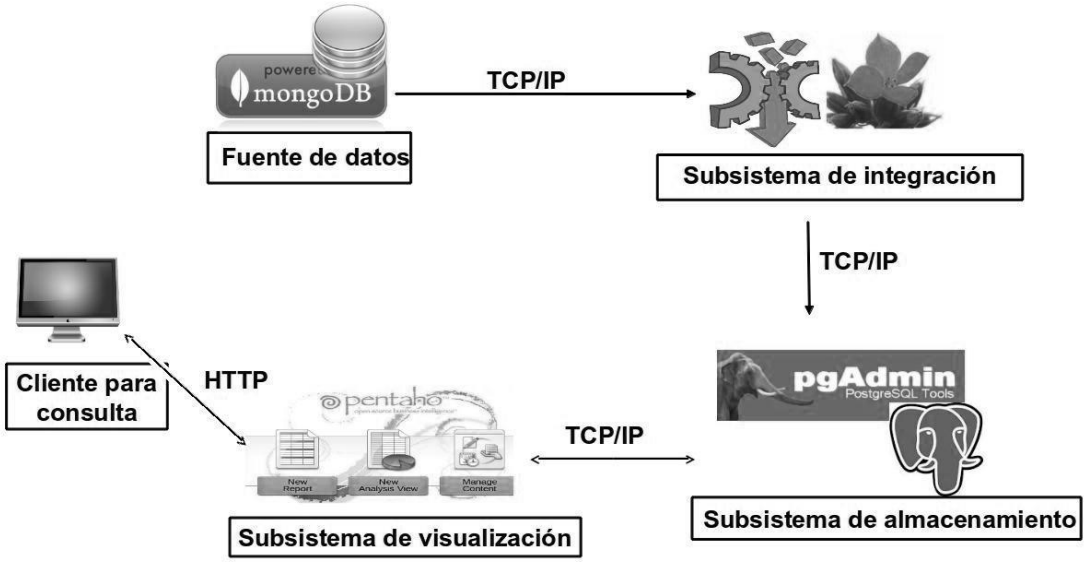


Figura 2: Arquitectura del sistema

2.6 Diseño del mercado de datos

2.6.1 Subsistema de almacenamiento

En el diseño del subsistema de almacenamiento se identifican las dimensiones y las tablas de hechos que son elementos imprescindibles para la realización del subsistema.

Dimensiones

Las dimensiones definidas para el mercado de datos son las siguientes:

1. Dimensión temporal (**dim_temporal**): esta dimensión describe el universo de valores bajo los que se puede clasificar la información atendiendo al tiempo.

Jerarquía: (dim_temporal_anno > dim_temporal_mes > dim_temporal_semana >

dim_temporal_dia)

2. Dimensión servidores (**dim_servidor**): esta dimensión describe el universo de valores bajo los cuales puede clasificarse la información atendiendo al servidor.

Jerarquía: (dim_servidor)

3. Dimensión descripción del CPU (**dim_descripción_cpu**): esta dimensión describe el universo de valores bajo los cuales puede clasificarse la información atendiendo a la descripción del CPU.

Jerarquía: (dim_descripcion_cpu)

4. Dimensión sistema operativo (**dim_descripción_so**): esta dimensión describe el universo de valores bajo los cuales puede clasificarse la información atendiendo al sistema operativo.

Jerarquía: (dim_descripcion_so)

5. Dimensión procesos (**dim_procesos**): esta dimensión describe el universo de valores bajo los cuales puede clasificarse la información atendiendo al tipo de proceso.

Jerarquía: (dim_procesos)

6. Dimensión postgres (**dim_postgres**): esta dimensión describe el universo de valores bajo los cuales puede clasificarse la información atendiendo al tipo de proceso de postgres.

Jerarquía: (dim_postgres)

Tabla de hechos

Los hechos identificados para el desarrollo del mercado son los siguientes:

1. Hecho rendimiento de los servidores **(hecho_rendimiento_servidores)**: este hecho recoge los datos referentes al rendimiento de los servidores de PostgreSQL.

2. Hecho descripción del CPU (**hecho_descripcion_cpu**): este hecho recoge los datos referentes a la descripción del CPU.

3. Hecho procesos más consumidores (**hecho_procesos_consumidores**): este hecho recoge los datos referentes a los procesos más consumidores.

4. Hecho procesos de postgres más consumidores (**hecho_postgres**): este hecho recoge los datos referentes a los procesos de postgres más consumidores.

Matriz bus

La matriz dimensional o matriz bus tiene como objetivo principal representar gráficamente las relaciones que existen entre la tabla de hechos y las dimensiones del mercado de datos, a continuación en la Tabla 5 se presenta dicha matriz:

Tabla 5: Matriz Bus

Hechos/dimensiones	Dim_ temporal	Dim_ servidor	Dim_ descrip_so	Dim_ descrip_cpu	Dim_ procesos	Dim_ postgres
Rendimiento_servidores	X	X	X			
Descripción_cpu	X	X	X	X		

Procesos_cnsumidores	X	X			X	
Postgres	X	X				X

Modelo de datos

Luego de realizada la definición de los hechos, dimensiones y medidas, se da paso a la estructuración del modelo de datos. Una de sus características principales es que no necesita una predefinición de los reportes, debido a que se diseñan de forma tal que cubra el universo de variantes que los usuarios necesiten consultar. A continuación en la Figura 3 se muestra el modelo de datos de la presente investigación, el cual muestra las relaciones existentes entre las tablas contenidas en el mercado de datos. Además muestra los dos esquemas que se definieron, uno es naire2 donde quedan recogidas las tablas de hechos y el otro se nombra dimensiones donde almacenan todas las dimensiones contenidas en el mercado de datos:

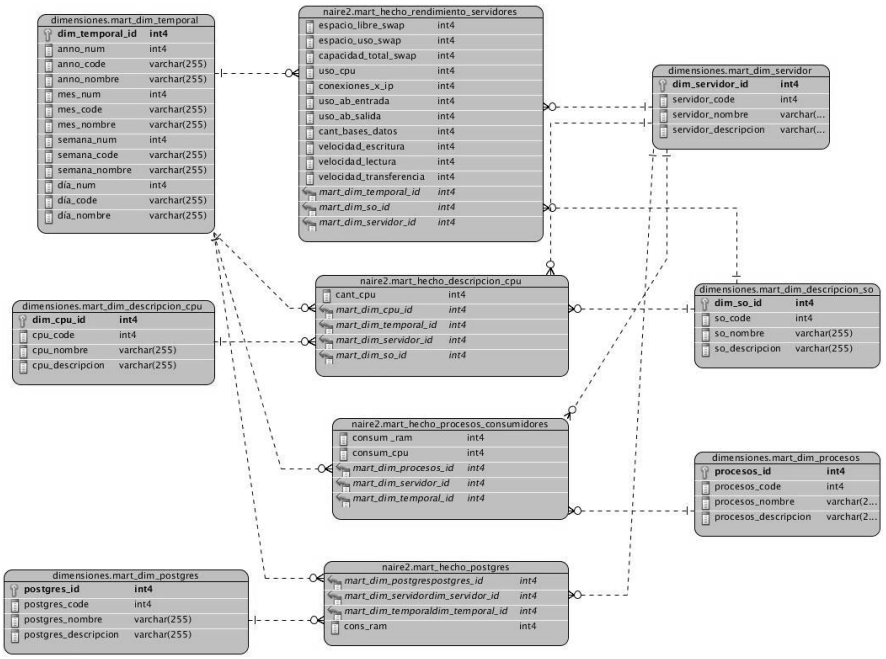


Figura 3: Modelo de datos

2.6.2 Subsistema de integración

En el diseño del subsistema de integración son elementos esenciales el perfilado de los datos así como el diseño de las transformaciones.

Perfilado de los datos

Con el perfilado de los datos se logra un mayor entendimiento de los datos, así como se verifica que no existan valores nulos, duplicados, distintos, de esta forma se definen nuevas reglas del negocio que tomarían como nombre reglas de transformación llevadas a cabo en la implementación del subsistema de integración. Luego de realizado el análisis de los resultados arrojados por el perfilado de los datos se concluye lo siguiente:

- Los tipos de datos son varchar y enteros.
- No existen valores nulos o vacíos.
- No existen valores negativos.

Diseño de los procesos de integración

Una vez analizados los datos se procede al diseño de las transformaciones. A la hora de la implementación estas transformaciones pueden sufrir cambios debido a disímiles situaciones que ocurren con los datos y para contrarrestar esto se llevan a cabo varias estrategias.

A continuación mediante esta transformación en la Figura 4 se muestra el procedimiento que se lleva a cabo a la hora de cargar los datos en el mercado de datos:

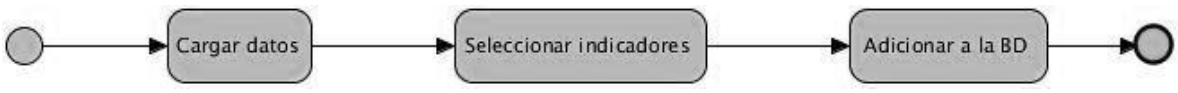


Figura 4: Diseño de subsistema de integración

En el caso de la carga de los hechos en el mercado de datos se lleva a cabo la siguiente transformación (ver Figura 5) donde se ponen de manifiesto los diferentes pasos por los que van pasando los datos, en este caso se inicia extrayendo los datos de la fuente, luego estos datos son validados, limpiados y transformados, luego se buscan los id de los indicadores, se validan estos id y en caso de existir algún error estos son enviados al Excel de errores y en caso de no existir ningún error se insertan los datos en la tabla de hechos:

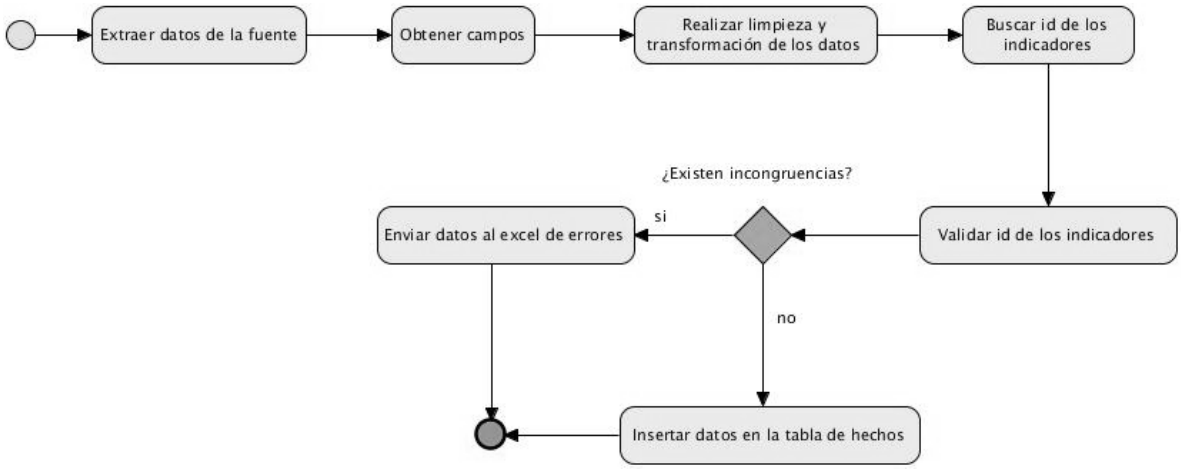


Figura 5: Diseño de subsistema de integración

2.6.3 Subsistema de visualización

Para la realización del diseño del subsistema de visualización son necesarios definir los reportes candidatos, así como los cubos OLAP.

El Área de análisis (AA) Servidores agrupa la información referente al indicador rendimiento de los servidores, la descripción del CPU y los procesos más consumidores de RAM y CPU, organizados por libros de trabajo.

LT Análisis del rendimiento de los servidores: El libro de trabajo contenido dentro del A.A Servidores. Contiene 14 reportes que permiten realizar un análisis general de los datos.

LT Análisis de la descripción del CPU: El libro de trabajo contenido dentro del A.A Servidores. Contiene 1 reporte que permite realizar un análisis general de los datos.

LT Análisis de los procesos más consumidores: El libro de trabajo contenido dentro del A.A Servidores. Contiene 4 reportes que permiten realizar un análisis general de los datos.

LT Análisis de los procesos de postgres más consumidores: El libro de trabajo contenido dentro del A.A Servidores. Contiene 1 reporte que permite realizar un análisis general de los datos.

Diseño de los cubos OLAP

En el diseño del subsistema de visualización es necesario crear los cubos OLAP (ver Figura 6). Estos proveen un mecanismo para buscar datos con rapidez, con el objetivo de agruparlos y proporcionar una vista más detallada, además, el tiempo de respuesta es uniforme, independientemente de la cantidad de datos.

En los cubos se definen las dimensiones, las medidas y los niveles jerárquicos de cada dimensión. A continuación se representa el diseño de los cubos dimensionales presentes en la aplicación:

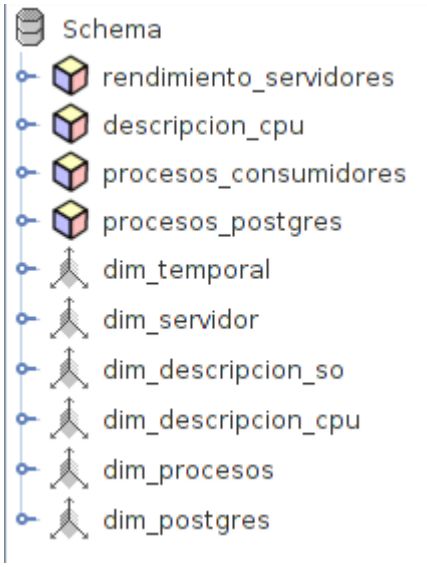


Figura 6: Cubos OLAP

2.7 Política de respaldo y recuperación

Con el objetivo de garantizar la persistencia de la información, se establece una política de respaldo y recuperación que comprende tres elementos esenciales:

• **Periodicidad de las salvas:** las salvas de toda la información contenida en la base de datos se realizarán con una periodicidad diaria, así lo define el departamento, verificando la existencia de una copia de toda la información almacenada.

• **Tablas involucradas:** se encuentra implicada 4 tablas de hechos identificadas en el proceso de análisis con sus 6 dimensiones asociadas.

• **Backups existentes:** en cada año se realizarán dos salvas, una en el mes de julio y la otra en el mes de diciembre. Se cuenta, además, con un servidor de respaldo con todos los datos y casos de ocurrencia de incidentes que atenten contra la seguridad de la entidad. También se tiene copia de los datos en otros medios de almacenamiento como DVD.

Conclusiones del capítulo

En el desarrollo del presente capítulo, se abordaron los principales elementos relacionados con los artefactos generados durante la etapa de análisis y diseño del mercado de datos.

✓ El levantamiento de requisitos dio lugar a la identificación de 15 requisitos de información, 21 requisitos funcionales, 7 requisitos no funcionales y 9 reglas del negocio.

✓ Se diseñó como parte del subsistema de almacenamiento el modelo de datos identificándose en el mismo la relación de las 4 tablas de hechos y las 6 tablas de dimensiones.

✓ Como parte del subsistema de implementación fueron diseñadas las transformaciones de forma general que servirá posteriormente para realizar el proceso de ETL.

✓ Se diseñó como parte del subsistema de visualización la arquitectura de la información identificada por un área de análisis, 4 libros de trabajo y 20 reportes. Se elaboró la política de respaldo y recuperación definiéndose así la periodicidad diaria de la información, para contribuir a la seguridad del mismo.

CAPÍTULO 3. IMPLEMENTACIÓN Y PRUEBAS

Introducción

En este capítulo se describen los procesos asociados a la fase de implementación y prueba del mercado de datos. Se abordarán aspectos referentes al transcurso de la solución para lograr su certificación. Se especificarán detalles acerca de la implementación de los subsistemas de integración y visualización de los datos. Se realizarán las pruebas pertinentes como parte de una actividad más para el aseguramiento de la calidad del producto, comprobando el correcto funcionamiento de la aplicación.

3.1 Implementación del subsistema de almacenamiento

Esquemas

Los esquemas no son más que formas de organización de la BD que pueden contener tablas, tipos de datos, funciones y operadores. A diferencia de la BD, los esquemas no están apartados entre sí, permitiéndole a un usuario acceder a cualquiera de ellos, si tiene los permisos adecuados.

El uso de esquemas trae consigo un grupo de mejoras, las cuales son mencionadas a continuación (25):

1. Organizan los objetos de la BD en grupos lógicos, facilitando su manejo.

2. Permiten el uso de la BD por parte de muchos usuarios sin que estos interfieran entre sí.

3. Permiten utilizar el mismo nombre para un objeto en esquemas diferentes sin ocasionar un conflicto en la BD.

4. Organizan los objetos de la BD en grupos lógicos, facilitando su manejo.

5. Permiten utilizar el mismo nombre para un objeto en esquemas diferentes sin ocasionar un conflicto en la BD.

Para el desarrollo del sistema propuesto en esta investigación se definieron dos esquemas:

- ✓ El esquema dimensiones: contiene las 6 tablas de dimensiones.

✓ El esquema naire2: contiene 3 tablas de hechos.

Tablas

A continuación en la Tabla 6 se muestra que la solución cuenta con 9 tablas en total, 6 dimensiones y 4 hechos, distribuidas por los dos esquemas propuestos con anterioridad quedando de la siguiente manera:

Tabla 6: Esquemas

Tablas	Esquemas	Descripción
Dim_temporal	dimensiones	Dimensión temporal
Dim_descripcion_cpu	dimensiones	Dimensión descripción CPU
Dim_postgres	dimensiones	Dimensión procesos postgres más consumidores de RAM y CPU
Dim_servidor	dimensiones	Dimensión servidor
Dim_descripción_so	dimensiones	Dimensión descripción del sistema operativo
Dim_procesos	dimensiones	Dimensión procesos más consumidores de RAM y CPU
Hech_rendimiento_servidores	naire2	Hecho rendimiento de los servidores
Hech_descripción_cpu	naire2	Hecho descripción de CPU

Hecho_procesos_consumidores	naire2	Hecho procesos más consumidores
Hecho_postgres	naire2	Hecho procesos de postgres más consumidores

Restricciones

Cuando se diseña una base de datos se debe reflejar fielmente el universo del discurso que se está tratando, o mejor, las restricciones existentes en el mundo real. Los componentes de una restricción son los siguientes (25):

✓ La operación de actualización (inserción, borrado o eliminación) cuya ejecución ha de dar lugar a la comprobación del cumplimiento de la restricción.

✓ La condición que debe cumplirse, la cual es en general una proposición lógica, definida sobre uno o varios elementos del esquema, que puede tomar uno de los valores de verdad (cierto o falso).

✓ La acción que debe llevarse a cabo dependiendo del resultado de la condición.

Las restricciones son condiciones que se le aplican a una BD para que cumpla con ciertos parámetros. Las mismas pueden ser creadas automáticamente al definir una tabla (en el caso de las llaves primarias) o ser introducidas por el programador de la BD cuando se busca algo específico. Existen 4 tipos de restricciones que son las más comunes:

✓ Clave foránea: son las que referencian una clave de otra tabla, se usan para relacionar tablas diferentes.

✓ De unicidad: implica que no debe haber 2 valores iguales en la misma columna.

✓ Clave primaria: son valores que deben cumplir con un conjunto de restricciones: no tener valores nulos, ser únicos para cada tupla y ser necesarios.

- ✓ Valor no nulo: no debe existir ninguna casilla de la columna que esté vacía.

En el mercado de datos en cada tabla de dimensiones el tipo de clave es primaria y para las tablas de hechos son foráneas.

3.2 Implementación del subsistema de integración

Implementación de las transformaciones

Las transformaciones constituyen un elemento básico dentro de la implementación del proceso de ETL. Dicho proceso consiste en extraer los datos de la fuente, limpiarlos y realizarles las transformaciones necesarias para finalmente cargarlas en el mercado de datos. En el mercado de datos para el apoyo a la toma de decisiones sobre los servidores de PostgreSQL se realizaron un total de 10 transformaciones, 6 se corresponden a las dimensiones y 4 a los hechos. Además se implementan los trabajos para organizar el orden de ejecución de las transformaciones.

A continuación en la Figura 7 se muestra la transformación del hecho descripción de CPU donde primeramente se obtienen los datos a través del componente de entrada en este caso la fuente de datos es Mongodb, luego se obtienen los campos con los que se va a trabajar, se le realiza la limpieza a los datos y se buscan en la base de datos el id de cada una de las dimensiones asociadas a este hecho y luego se procede a la carga del mismo.

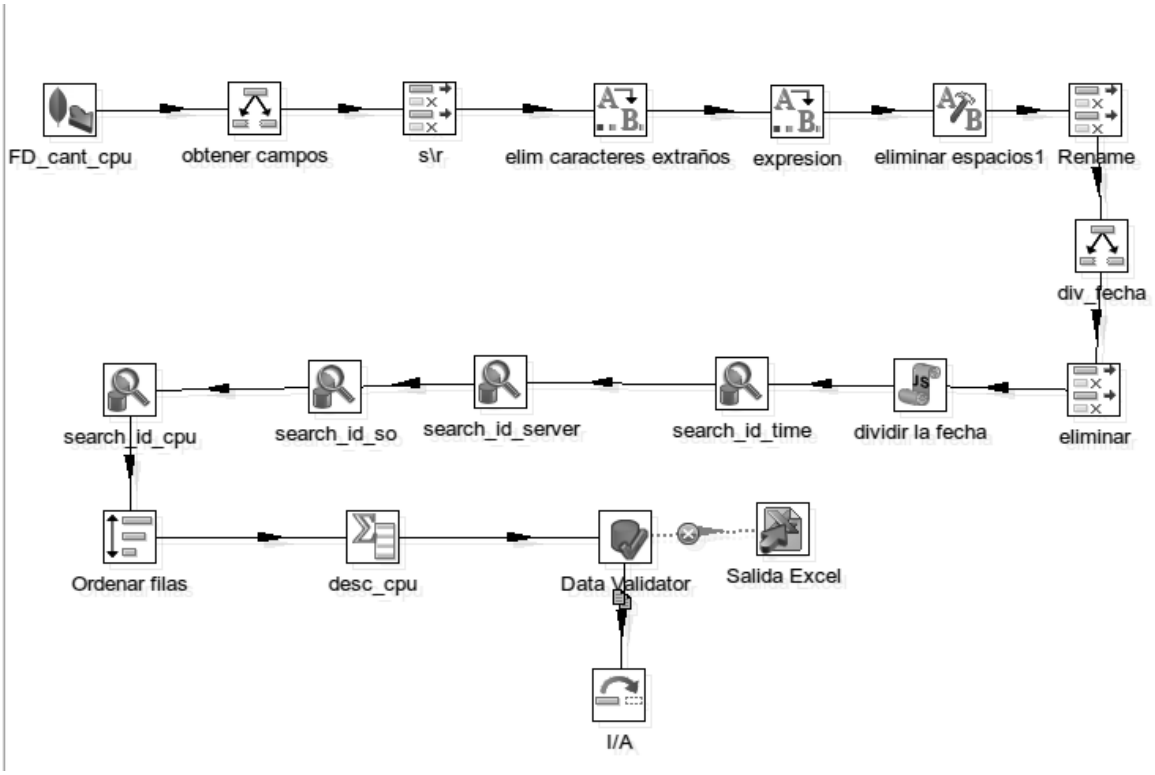


Figura 7: Transformación del hecho descripción de CPU

Implementación de los trabajos

Al concluir la realización de todas las transformaciones necesarias para la carga de los datos, se realiza la implementación de los trabajos, el cual se encarga de ejecutar todas las transformaciones en un orden lógico definido, primero las dimensiones y después los hechos.

A continuación en la Figura 8 se muestra el trabajo para cargar las dimensiones presentes en el mercado de datos.

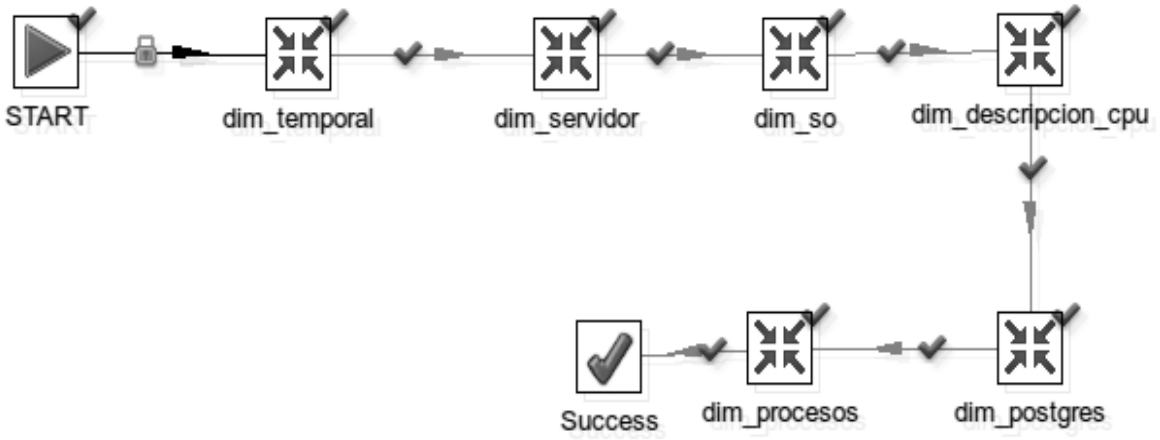


Figura 8: Trabajo para cargar las dimensiones

3.3 Implementación del subsistema de visualización

Implementación de los cubos OLAP

Para la implementación de los cubos, primero se definieron las dimensiones compartidas entre ellos, y luego se implementó un cubo por cada una de las tablas de hecho. La Figura 9 muestra la implementación del cubo para el hecho rendimiento de servidores.

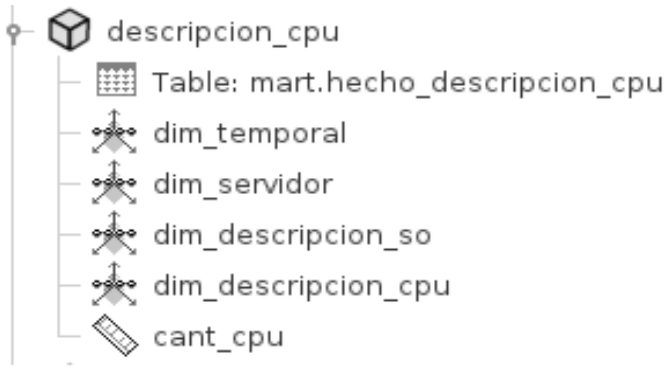


Figura 9: Cubo rendimiento de servidores

Implementación de los reportes

Los reportes candidatos o tablas de salidas como también se les conoce van a contener los valores asociados a los indicadores que son de interés para el cliente, estos son implementados a través de consultas MDX que son en los sistemas OLAP el equivalente a las consultas SQL en las bases de datos relacionales. A continuación en la Figura 10 se muestra una vista de análisis implementada a través de consultas MDX.

Select NON EMPTY {[Measures]. [cant_cpu]} ON COLUMNS, NON EMPTY Crossjoin ([dim_temporal]. [2013]. Children, Crossjoin ([dim_servidor]. [Todas].Children, Crossjoin ([dim_descripcion_so]. [Todas].Children, [dim_descripcion_cpu]. [Todos].Children))) **ON ROWS from** [descripcion_cpu]

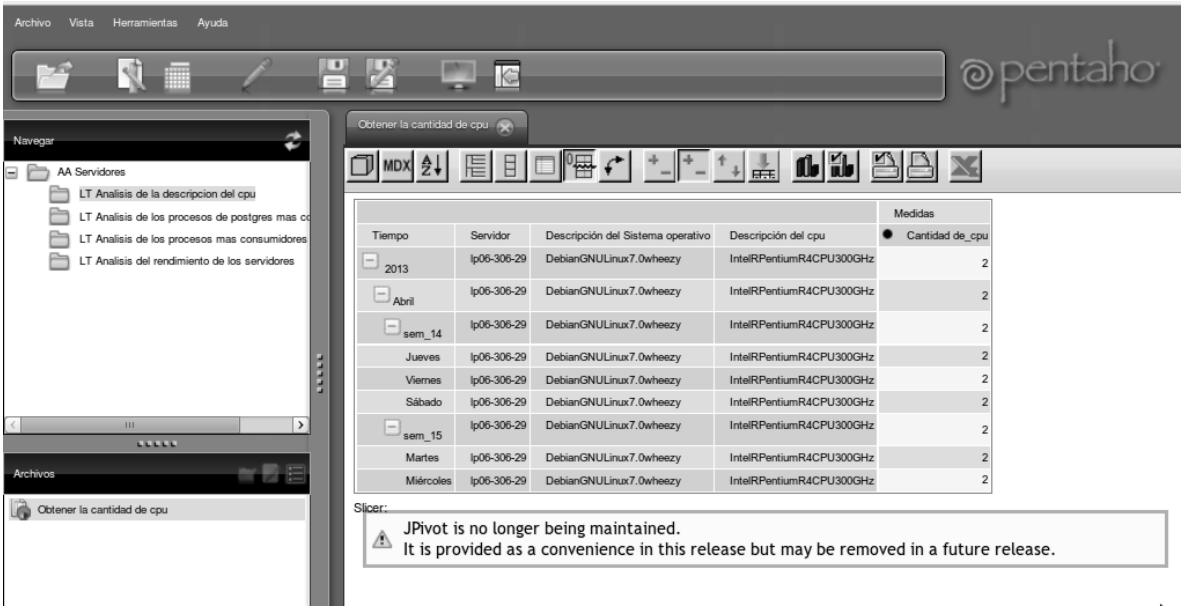


Figura 10: Vista de análisis de la cantidad de CPU

3.4 Esquema de seguridad

Resulta de gran importancia para un sistema de información el hecho de contar con un mecanismo de protección contra aquellas acciones que puedan afectar la integridad de los datos almacenados. Por tal

motivo, para el acceso al mercado de datos se ha establecido un usuario por cada uno de los roles existentes en el sistema, con el objetivo de garantizar un control de acceso. De esta manera cada usuario opera en el sistema según los permisos que se le definan al rol correspondiente.

Roles y permisos

Debido al intenso análisis del negocio realizado para definir quiénes son los que tienen acceso a la información contenida dentro de la BD que se implementará, se define el rol“ Administrador ETL”, y para la seguridad en la aplicación se definen dos roles “Analista” y ”Administrador”.

- **Seguridad en la base de datos**

Administrador ETL: Tiene acceso total a todas las Áreas de Análisis (A.A) del mercado de datos. Realiza los procesos de ETL sobre los datos.

- **Seguridad en la aplicación**

Analista: Tiene acceso de solo lectura al A.A Servidores. Puede realizar operaciones sobre todos los reportes de esta área.

Administrador: Tiene acceso total para gestionar los permisos, roles y usuarios, pero solo tiene permiso de lectura en las A.A del mercado de datos.

3.5 Pruebas

Para los ingenieros del software desarrollar un producto o servicio de buena calidad y aceptación por el cliente, constituye un requisito indispensable para que el propio ingeniero pueda ganar determinada reputación dentro de la entidad donde se desempeña como profesional y también la empresa u organización a la que pertenecen. Muchos son los autores que han sido capaces de hablar de diferentes conceptos relacionados con la calidad de los productos de software. Uno de los conceptos más íntegros y fácil de comprender por el lector es el que a continuación se muestra.

La calidad del software es el grado con el que un sistema, componente o proceso cumple los requerimientos específicos y las necesidades o expectativas del cliente o usuario. (26)

Durante el desarrollo de un producto de software la forma de evitar que los errores se produzcan y, sobre todo que se propaguen es disponer de procedimientos de calidad y pruebas que acompañen al producto a lo largo de su ciclo de vida.

En la presente investigación para evaluar el mercado de datos se utilizó el Modelo V (ver Figura 11), que no es más que una adaptación de los procesos de validación y verificación sobre el proceso de desarrollo. El modelo V constituye una evolución del modelo en cascada, con la diferencia que este considera que las actividades de prueba se ejecuten paralelamente con las actividades de análisis y diseño. Representa las etapas del ciclo de desarrollo de las pruebas con un nivel de abstracción elevado. Este modelo hace corresponder con cada etapa de desarrollo una prueba pertinente en su nodo paralelo de la columna de aseguramiento de la calidad. Cada una de las pruebas se encarga de prevenir futuros desperfectos en los requisitos, diseño e implementación del sistema.

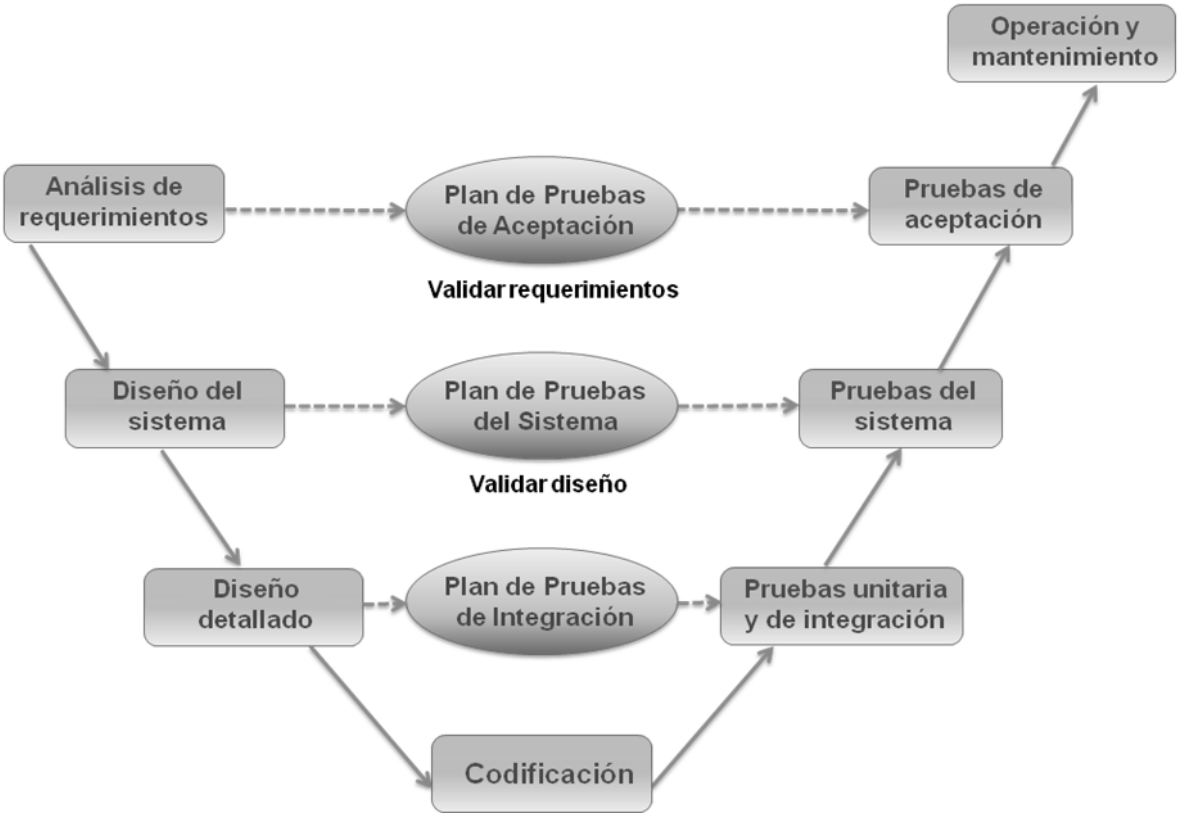


Figura 11: Etapas que comprueba el modelo V

3.6 Herramientas para evaluar las pruebas

Casos de prueba

Los casos de prueba permiten verificar que el producto desarrollado satisface los requerimientos del usuario, tal y como se describe en la especificación de requerimientos y casos de uso.

Para lograr la calidad del producto de software es necesario realizar un conjunto de evaluaciones durante todo el proceso de desarrollo que impliquen al cliente y desarrollador. Para ello se diseñaron casos de prueba basados en los casos de uso y a su vez en los requerimientos funcionales,

comparando cada funcionalidad implementada con la descrita, para verificar hasta qué punto cumplía con las necesidades del cliente. (27)

Para el Mercado de datos para el apoyo a la toma de decisiones sobre servidores de PostgreSQL se diseñaron un total de 9 casos de prueba, a continuación se muestra una imagen (ver Figura 12) donde se presenta un escenario del caso de uso de información Rendimiento de los servidores:

Escenario	Descripción	Variables		Respuesta del sistema	Flujo central
		Perfiles de análisis	Indicadores a medir		
EC 1.1: Análisis rendimiento servidores	Permite visualizar el reporte con las variables presentes en el mismo.	Año Servidor Descripción sistema operativo	espacio_libre_swap espacio_uso_swap capacidad_total_swap uso_cpu conexiones_por_ip uso_ancho_banda_entrada uso_ancho_banda_salidad cant_base_datos veloc_escritura veloc_lectura veloc_transferencia	Se muestra la tabla con los valores correspondientes a cada escenario.	1. El usuario se autentica en el sistema 2. Selecciona el AA Servidores 3. Se selecciona el LT Análisis del rendimiento de los servidores 4. En el área de trabajo se visualiza la tabla que corresponde al reporte

Figura 12: Escenario del caso de uso rendimiento de los servidores

Listas de chequeo

Las listas de chequeo son un conjunto de preguntas que permiten comprobar el cumplimiento de los objetivos planteados. En el mercado de datos para el apoyo a la toma de decisiones sobre servidores de PostgreSQL la lista de chequeo utilizada (ver Tabla 7) se divide en tres secciones:

- **Estructura del documento:** Contiene todos los aspectos definidos por el expediente del proyecto.

- **Indicadores definidos en el desarrollo:** Contiene todos los indicadores a evaluar durante la etapa de análisis de los datos.

- **Semántica del documento:** Contiene todos los indicadores a evaluar respecto a la redacción y ortografía.

Estructura de la lista de chequeo

- **Peso:** define si el indicador a evaluar es crítico o no.
- **Indicadores a evaluar:** son los indicadores a evaluar en las secciones Estructura del documento, semántica del documento e indicadores definidos por las diferentes etapas.
- **Evaluación:** es la forma de evaluar el indicador en cuestión. El mismo se evalúa de 1 en caso de que exista alguna dificultad sobre el indicador y 0 en caso de que el indicador revisado no presente problemas.
- **No procede:** se usa para especificar que el indicador no es necesario evaluarlo en ese caso.
- **Cantidad de elementos afectados:** especifica la cantidad de errores encontrados sobre el mismo indicador.

Tabla 7: Lista de chequeo

Estructura del documento					
Peso	**Indicadores a evaluar**	**Eval**	**(NP)**	**Cantidad de elementos afectados**	**Comentarios**
crítico	1. ¿Están los documentos acorde a las planillas estándar definidas por el proyecto?	0		0	

crítico	2. ¿Contiene las secciones obligatorias definidas en el expediente? (Ver Expediente de Proyecto del Departamento)	0		0	
Elementos definidos por el modelo de desarrollo					
Peso	**Indicadores a evaluar**	**Eval**	**(NP)**	**Cantidad de elementos afectados**	**Comentarios**
	1. ¿Se realizaron estudios preliminares de la entidad cliente?	0		0	
crítico	2. ¿Se identificaron las necesidades de información y las reglas del negocio?	0		0	
crítico	3. ¿Se realizó el diseño del modelo de datos correspondiente al Mercado de Datos para el apoyo a la toma de decisiones sobre los servidores de PostgreSQL en conjunto con el cliente y los especialistas del centro?	0		0	
	4. ¿Se realizaron los diseños de los procesos Extracción, Transformación y Carga para el Mercado de Datos para el apoyo a la toma de decisiones sobre los servidores de PostgreSQL?	0		0	
crítico	5. ¿Se realizaron los procesos de	0		0	

	Extracción, Transformación y Carga correspondiente al Mercado de Datos para el apoyo a la toma de decisiones sobre los servidores de PostgreSQL?				
	6. ¿Las transformaciones se pueden ejecutar desde cualquier PC?	0		0	
crítico	7. ¿Se realizaron las implementaciones de los trabajos para el Mercado de Datos para el apoyo a la toma de decisiones sobre los servidores de PostgreSQL?	0		0	
crítico	8. ¿Se le dan tratamiento a los errores que ocurren en el proceso de Extracción, Transformación y Carga?	0		0	
crítico	9. ¿Se realizó el proceso de Inteligencia de Negocio correspondiente al Mercado de Datos para el apoyo a la toma de decisiones sobre los servidores de PostgreSQL?	0		0	
crítico	10. ¿Los reportes que se muestran en la capa de visualización se corresponden con las necesidades del negocio identificadas?	0		0	
crítico	11. ¿La aplicación realizada apoya el proceso de toma de decisiones para las áreas de los Servidores?	0		0	
crítico	12. ¿Se realizaron los diseños de los casos de prueba?	0		0	

crítico	13. ¿Se realizaron las Pruebas de unidad?	0		0	
crítico	14. ¿Se realizaron las Pruebas de aceptación?	0		0	
crítico	15. ¿Se realizaron las Pruebas de integración?				
crítico	16. ¿Se realizaron las Pruebas de sistema?				
crítico	17. ¿Se realizó el despliegue de la aplicación?	0		0	
	18. ¿Se realizaron las Pruebas pilotos?	0		0	
crítico	19. ¿Se le da soporte y mantenimiento a la aplicación?	0		0	
Semántica del documento					
Peso	**Indicadores a evaluar**	**Eval**	**(NP)**	**Cantidad de elementos afectados**	**Comentarios**
crítico	1. ¿Se han identificado errores ortográficos en los entregables?	1		1	
crítico	2. ¿Se entiende claramente lo que se ha especificado en el documento?	0		0	
	3. ¿El número de página que aparece en el índice coincide con el	0		0	

	contenido que se refleja realmente en dicha página?				

3.7 Resultado de las pruebas

Pruebas unitarias

Las pruebas unitarias se enfocan en un programa o componente que desempeña una función específica, verificando que funcione tal y como lo define la especificación del programa. Los programadores siempre prueban el código durante el desarrollo, por lo que las pruebas unitarias son realizadas solamente después de que el programador considera que el componente se encuentra libre de errores.

Una vez concluida la etapa de implementación se realizaron las pruebas unitarias al mercado de datos, donde se detectaron dos no conformidades, las cuales fueron resueltas satisfactoriamente en un corto período de tiempo.

Pruebas de integración

Su objetivo es identificar los errores introducidos por la combinación de programas o componentes probados unitariamente, además, verificar que las especificaciones de diseño sean alcanzadas. No son verdaderamente pruebas de sistema debido a que los componentes no se encuentran implementados en el ambiente operativo. Al sistema se le realizaron pruebas de integración donde no se detectaron no conformidades, al tener un correcto funcionamiento el subsistema de integración con el subsistema de visualización.

Pruebas del sistema

Son usualmente conducidas para asegurar que todos los módulos trabajan como sistema, sin error. Es similar a la prueba de integración pero con un alcance mucho más amplio. Las pruebas del sistema examinan qué tan bien el sistema cumple con los requerimientos de la organización y su utilidad, seguridad y desempeño. También se realizan estas pruebas a la documentación del sistema. Las pruebas del sistema fueron aplicadas por el grupo de calidad interna del departamento, a través de los casos de

prueba guiados por casos de uso. Los nueve casos de prueba basados en CU permitieron identificar en una primera iteración un total de 5 no conformidades, en una segunda iteración se detectaron 2 no conformidades relacionadas con la estructura de los casos de prueba, y los reportes candidatos, las cuales fueron totalmente resueltas.

Pruebas ejecutadas aplicando listas de chequeo

Luego de aplicada la lista de chequeo general a los principales artefactos se detectaron no conformidades que fueron resueltas en un corto período de tiempo. El siguiente gráfico (ver Figura 13) muestra el comportamiento de los indicadores definidos para la lista de chequeo. De forma general se identificaron 22 indicadores, de ellos 21 críticos y luego de aplicada la lista de chequeo se encontró una no conformidad:

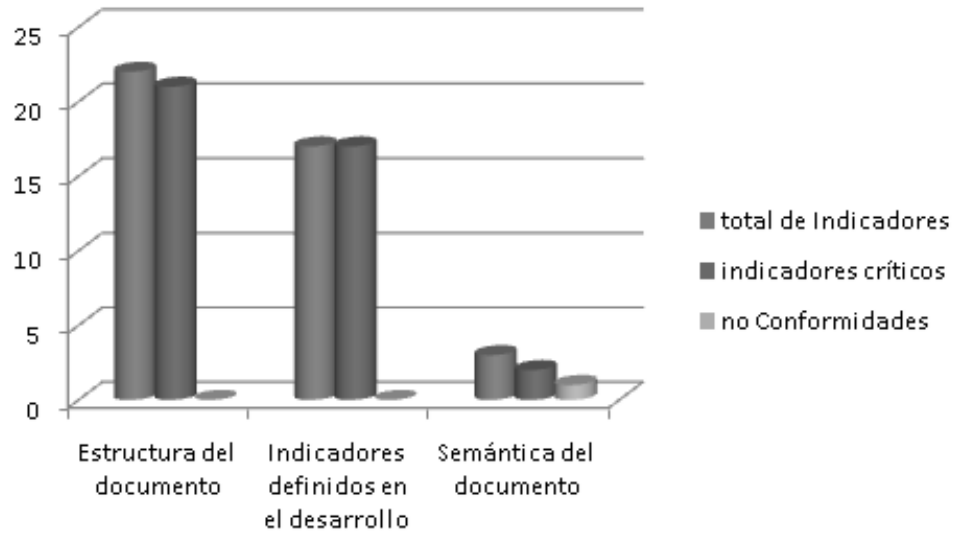


Figura 13: Indicadores listas de chequeo

Pruebas de aceptación

Las pruebas de aceptación son realizadas con el objetivo de determinar el grado de satisfacción del cliente con el sistema. También se confirma que el sistema está terminado y cumple con los requisitos definidos. Para verificar el estado de cumplimiento de la aplicación se realizó una entrevista con el

cliente, donde se le mostró la aplicación y quedó conforme con la misma, firmando una carta de aceptación.

Conclusiones del capítulo

En este capítulo se implementaron los subsistemas presentes en la arquitectura del sistema.

✓ En el caso del subsistema de almacenamiento quedaron definidos 2 esquemas uno para los hechos y otro para las dimensiones propios del mercado.

✓ En el subsistema de integración se implementaron los trabajos que permitieron ejecutar las transformaciones en un orden lógico definido.

✓ En el subsistema de visualización se implementaron los cubos OLAP, se realizó además la implementación de los reportes candidatos permitiendo así la visualización de los datos para el apoyo a la toma de decisiones.

✓ Para la evaluación del mercado de datos se utilizó el modelo V, se realizaron pruebas unitarias donde se encontraron dos no conformidades, pruebas de integración donde no se encontraron no conformidades y las pruebas al sistema donde se encontraron en una primera iteración cinco no conformidades y en la segunda dos no conformidades. Se utilizaron las listas de chequeo y los casos de prueba donde se verificó el cumplimiento de los objetivos y requisitos planteados, además se realizó la aceptación del mercado de datos por parte del cliente.

CONCLUSIONES GENERALES

Al finalizar el trabajo de diploma “Mercado de datos para el apoyo a la toma de decisiones sobre servidores PostgreSQL” se concluye que se ha cumplido con los objetivos específicos planteados. Para ello:

✓ La selección de la Propuesta de Metodología para el Desarrollo de Almacenes de Datos en DATEC garantizó el proceso de desarrollo del software. Las herramientas utilizadas permitieron desarrollar satisfactoriamente la aplicación y darle respuestas a las necesidades de información del cliente.

✓ Se realizó el análisis y diseño del mercado de datos donde se identificaron 4 tablas de hechos, 14 medidas y 6 dimensiones sirviendo como base para la fase de implementación. Además que se definieron los requisitos de información, funcionales y no funcionales logrando así cumplir con las necesidades del cliente.

✓ Se implementó el mercado de datos cumpliendo con los requerimientos definidos por el cliente. Se realizó el diseño de los cubos, se identificó el área de análisis así como los libros de trabajos contenidos dentro de esta.

✓ La aplicación de las pruebas al mercado de datos, permitió identificar 10 no conformidades que fueron resueltas satisfactoriamente.

RECOMENDACIONES

Con el objetivo de futuras mejoras para este trabajo se sugiere:

✓ Utilizar la presente investigación como base para futuros mercados de datos, según surjan nuevas métricas que contribuyan a mejorar el proceso de toma de decisiones sobre el rendimiento de los servidores de PostgreSQL.

REFERENCIAS BIBLIOGRÁFICAS

1. **Lianet Lores Sánchez, Diana Monné Roque.** Aplicación de las pruebas de liberación al Sistema Informático de Genética Médica. Trabajo de Diploma para optar por el título de Ingeniero Informático. [En línea] junio de 2009. [Citado el: 05 de 03 de 2013.]

2. **Lamancha, Beatriz Pérez.** Proceso de Testing funcional independientemente. Tesis de Maestría. [En línea] septiembre de 2006. [Citado el: 25 de 04 de 2013.]

3. **Juan Bernardo Quintero, Quispe-Otazu, Rodolfo.** ¿Que es la Calidad de Software? Blog de Rodolfo Quispe-Otazu. [En línea] 26 de julio de 2008. [Citado el: 29 de 04 de 2013.] http://www.rodolfoquispe.org/blog/que-es-la-calidad-de-software.php.

4. **OCHOA, Darián GONZALEZ.** Diseño e Implementación de un Almacén de Datos Operacionales para la corporación CIMEX. Tesis (Ingeniero en Ciencias Informáticas). Ciudad de La Habana. [En línea] 2009. [Citado el: 28 de 03 de 2013.]

5. inqbation. [En línea] [Citado el: 25 de 04 de 2013.] http://www.inqbation.com/pentaho-data-integration/..

6. **PÉREZ, Ma TERESA GARZÓN.***SISTEMAS GESTORES DE BASES DE DATOS.* 30 MAYO 2010.

7. **Masip, David.** http://www.desarrolloweb.com/articulos/840.php. [En línea] 19 de julio de 2002. http://www.desarrolloweb.com/articulos/840.php.

8. **Mora, Luis Alberto Casillas Santillán Marc Gibert Ginestà Óscar Pérez.***Base de datos en MySql.*

9. Global Development Group. PostgreSQL. [En línea] 22 de 11 de 2010. http://www.postgresql.org/....

10. **Inmon, W. H.***Building the Data Warehouse.* s.l. : Wiley Publishing,, 2005.

11. **Ross, Kimball &.***The Data Warehouse Toolkit: the Complete Guide to Dimensional Modelling.* New York, EE.UU: : s.n., 2002.

12. **Mario Roberto Reyes Marroquín, Pablo Augusto Rosales Tejada.***DESARROLLO DE UN DATAMART DE INFORMACIÓN ACADÉMICA DE ESTUDIANTES DE LA ESCUELA DE CIENCIAS.* Guatemala: s.n., : s.n., 2007.

13. **Ortiz, Marta Cecilia.***La inteligencia de negocios aplicada a las organizaciones en Latinoamérica.* 2007.

14. **Inmon, Bill.***Business Intelligence Galicia.* 2007.

15. **LUJAN MORA, Sergio.***Data Warehouse Design with UML Tesis (Doctorado). España, Universidad de Alicante,.* 2005.

16. **Berríos, Ericka Graciela Sevilla.** http://www.scribd.com. [En línea] Marzo de 2003. http://www.scribd.com/doc/39978465/GUIA-METODOLOGICA-PARA-LA-DEFINICION-Y- DESARROLLO-DE-UN-DATAWAREHOUSE..

17. Sinnexus. [En línea] 2007. [Citado el: 02 de octubre de 2012.] www.sinnexus.com..

18. **PostgreSQL Tools. PostgreSQL Tools. [En línea] [Citado el: 10 de 12 de 2012.]**

http://www.pgadmin.org.

19. sinnexus.sinnexus. [En línea] http://www.sinnexus.com/business_intelligence/olap_vs_oltp.aspx..

20. **López., Jorge Luis Tufiño.** http://bibdigital.epn.edu.ec/handle/15000/4101. [En línea] 09 de 2011. [Citado el: 05 de 12 de 2012.] http://bibdigital.epn.edu.ec/handle/15000/4101..

21. **Alberto Límia Navarro, Anisley Delfino, Asnioby Hernández López, Doris Medina Mustelier,.***Metodología para el Desarrollo de Soluciones de Almacenes de Datos e Inteligencia de Negocios en DATEC.* Ciudad de La Habana: : s.n., 2010.

22. Visual_Paradigm. Visual_Paradigm. [En línea] [Citado el: 28 de 11 de 2012.] http://www.ecured.cu/index.php/Visual_Paradigm..

23. IBM, International Business Machines. [En línea] 2010. http://www.ibm.com/es/es/..

24. UBUNTU. guia-ubuntu. PgAdmin_III, 2008. [En línea] [Citado el: 17 de 11 de 2012.] http://www.guia-ubuntu.org/indeX.php?title=PgAdmin_III..

25. http://www.bajaki.com/download/datacleaner.htm. [En línea] [Citado el: 15 de 03 de 2013.] http://www.bajaki.com/download/datacleaner.htm..

26. MSDN., Microsoft. Microsoft Developer Network. [En línea] 2010. http://msdn.microsoft.com/es-es/default.aspX..

27. mondrian.pentaho. [En línea] http://mondrian.pentaho.com/documentation/workbench.php..

BIBLIOGRAFÍA

Álvaro Moreno Aguilera, Carlos Muñoz García, Alberto Alvarado Flores.Herramientas ETL de código abierto. . [En línea] 2008. [En línea]

AUTORES, COLECTIVO DE. Metodología para el desarrollo de soluciones de. [En línea]

Berríos, Ericka Graciela Sevilla. GUIA-METODOLOGICA-PARA-LA-DEFINICION-Y-. [En línea]

(Bill Inmon, 2003 - Business Intelligence Galicia, 2007).

CAVSI. ¿Qué es un Sistema Gestor de Bases de Datos o SGBD? Disponible en:http://www.cavsi.com/preguntasrespuestas/que-es-un-sistema-gestor-de-bases-de-datos-o-. [En línea]

Chaudhuri, S. y Dayal. An Overview of Data Warehousing and OLAP Technology. [En línea]

Domínguez, Candelaria Yurena Ávila. Almacen de datos (DATA WAREHOUSE). [En línea]

Ecured. [En línea] http://www.ecured.cu/index.php/Sistema_Gestor_de_Base_de_Datos.

http://danielpecos.com/docs/mysql_postgres/x57.html mysql. [En línea]

http://www2.rhernando.net/modules/tutorials/doc/bd/oracle.html. [En línea]

http://www2.rhernando.net/modules/tutorials/doc/bd/oracle.html oracle. [En línea]

Intelligence., Pentaho Corporation. Pentaho Open Source Business. http://www.pentaho.com/. [En línea] 2010. [En línea]

Inmon, W. H. Building the Data Warehouse. S.l.: Wiley Publishing, 2005.

Intelligence., Pentaho Corporation. Pentaho Open Source Business. http://www.pentaho.com/. [En línea] 2010. [En línea]

MSDN., Microsoft. Microsoft Developer Network. http://msdn.microsoft.com/es-es/default.aspX.[En línea] 2010. [En línea]

Mysql http://www.google.com.cu/url?sa=t&rct=j&q=sgbd+mysql&source=web&cd=10&ved=0CGEQFjAJ&url=http%3A%2F%2Focw.uoc.edu%2Fcomputer-science-technology-and-multimedia%2Fbases-de-datos%2Fbases-de-datos%2FP06_M2109_02151.pdf&ei=sTKZUIimIqb40gGk8oD4Aw&usg=AFQjC. [En línea]

OCHOA, Darián GONZALEZ. Diseño e Implementación de un Almacén de Datos Operacionales para la corporación CIMEX. Tesis (Ingeniero en Ciencias Informáticas). Ciudad de La Habana. [En línea]

Open Source Business Intelligence.Portada sobre la plataforma Pentaho. Portada sobr. [En línea]

Ortiz, Marta Cecilia Ortiz. La inteligencia de negocios aplicada a las organizaciones en Latinoamérica. [En línea] 2007 Mercado de datos.

POstgreSQl 3. Global Development Group. PostgreSQL. . [En línea] 22 de 11 de 2010 . http://www.postgresql.org/.. [En línea]

Ross, Kimball &. The Data Warehouse Toolkit: the Complete Guide to Dimensional Modelling. New York, EE.UU: John Wiley & Sons, 2002. [En línea]

Ross, Kimball &. The Data Warehouse Toolkit: the Complete Guide to Dimensional Modelling. New York, EE.UU: John Wiley & Sons, 2002. [En línea]

Servidores http://www.um.es/docencia/barzana/DIVULGACION/INFORMATICA/sgbd.html. [En línea]

SINNEXUS. ¿Qué es Business Intelligence? [Consultado el: 12/11/2013 Disponible en:. [En línea]

SINNEXUS. Persistencia MOLAP,ROLAP,HOLAP Disponible en: http://www.sinnexus.com/business_intelligence/olap_avanzado.aspx. [En línea]

SINNEXUS.Datawarehouse Disponible en: http://www.sinnexus.com/business_intelligence/datawarehouse.aspx. [En línea]

sinnexus.sinnexus.[En línea] [Citado http://www.sinnexus.com/business_intelligence/olap_vs_oltp.aspx. [En línea]

sinnexus.sinnexus.[En línea] [Citado http://www.sinnexus.com/business_intelligence/olap_vs_oltp.aspx. [En línea]

Wood, Sherman. Pentaho Mondrian Project. . [En línea] 2008. [En línea]

ANEXOS

String analyzer (Fecha,ServerName,descrip,release)

	Fecha	ServerName	descrip	release
Row count	33733	33733	33733	33733
Null count	0	0	0	0
Blank count	0	0	0	0
Entirely uppercase count	0	0	0	0
Entirely lowercase count	0	33733	0	0
Total char count	640927	371063	978257	101199
Max chars	19	11	29	3
Min chars	19	11	29	3
Avg chars	19	11	29	3
Max white spaces	1	0	3	0
Min white spaces	1	0	3	0
Avg white spaces	1	0	3	0
Uppercase chars	0	0	168665	0
Uppercase chars (excl. first letters)	0	0	134932	0
Lowercase chars	0	67466	505995	0
Digit chars	472262	236131	67466	67466
Diacritic chars	0	0	0	0
Non-letter chars	640927	303597	303597	101199
Word count	67466	33733	134932	33733
Max words	2	1	4	1
Min words	2	1	4	1

Anexo 1: Perfilado de datos para los campos cadenas

Number analyzer (libre,total,usada)

	libre	total	usada
Row count	33734	33734	33734
Null count	0	0	0
Highest value	952	952	213
Lowest value	739	952	0
Sum	32.100.988	32.114.768	13.780
Mean	951,592	952	0,408
Geometric mean	951,555	952	0
Standard deviation	7,901	0	7,901
Variance	62,422	0	62,422

Anexo 2: Perfilado de datos a los campos numéricos

GLOSARIO DE TÉRMINOS

CALISOFT: Centro de Calidad para Aplicaciones Tecnológicas.

BD: Bases de Datos

BI: Inteligencia de Negocio

Data Warehouse: Almacén de Datos

DATEC: Centro de Tecnologías de Gestión de Datos

ETL: Extracción, Transformación y Carga

Granularidad: representa el nivel de detalle al que se desea almacenar la información y se define en dependencia del negocio que se esté analizando.

HOLAP: Procesamiento Analítico en Línea Híbrido

Json: Acrónimo de JavaScript Object Notation, es un formato ligero para el intercambio de datos es un subconjunto de la notación literal de objetos de JavaScript que no requiere el uso de XML.

MD: Mercado de Datos

MOLAP: Procesamiento Analítico en Línea Multidimensional

MongoDB: el nombre viene del término inglés “humongous” que significa colosal. Es una base datos no relacionales, es decir no utiliza SQL orientada a documentos json.

No conformidad: defecto, error o sugerencia que se le hace al equipo de desarrollo una vez encontrada alguna dificultad en lo que se está evaluando.

SGBD: Sistema Gestor de Bases de Datos

OLAP: Procesamiento Analítico en Línea

ROLAP: Procesamiento Analítico en Línea Relacional

UCI: Universidad de las Ciencias Informáticas